https://cnfl.io/ksql-workshop-prereq

Real-time SQL stream processing at scale with Apache Kafka® and KSQL

@rmoff

#stratadata

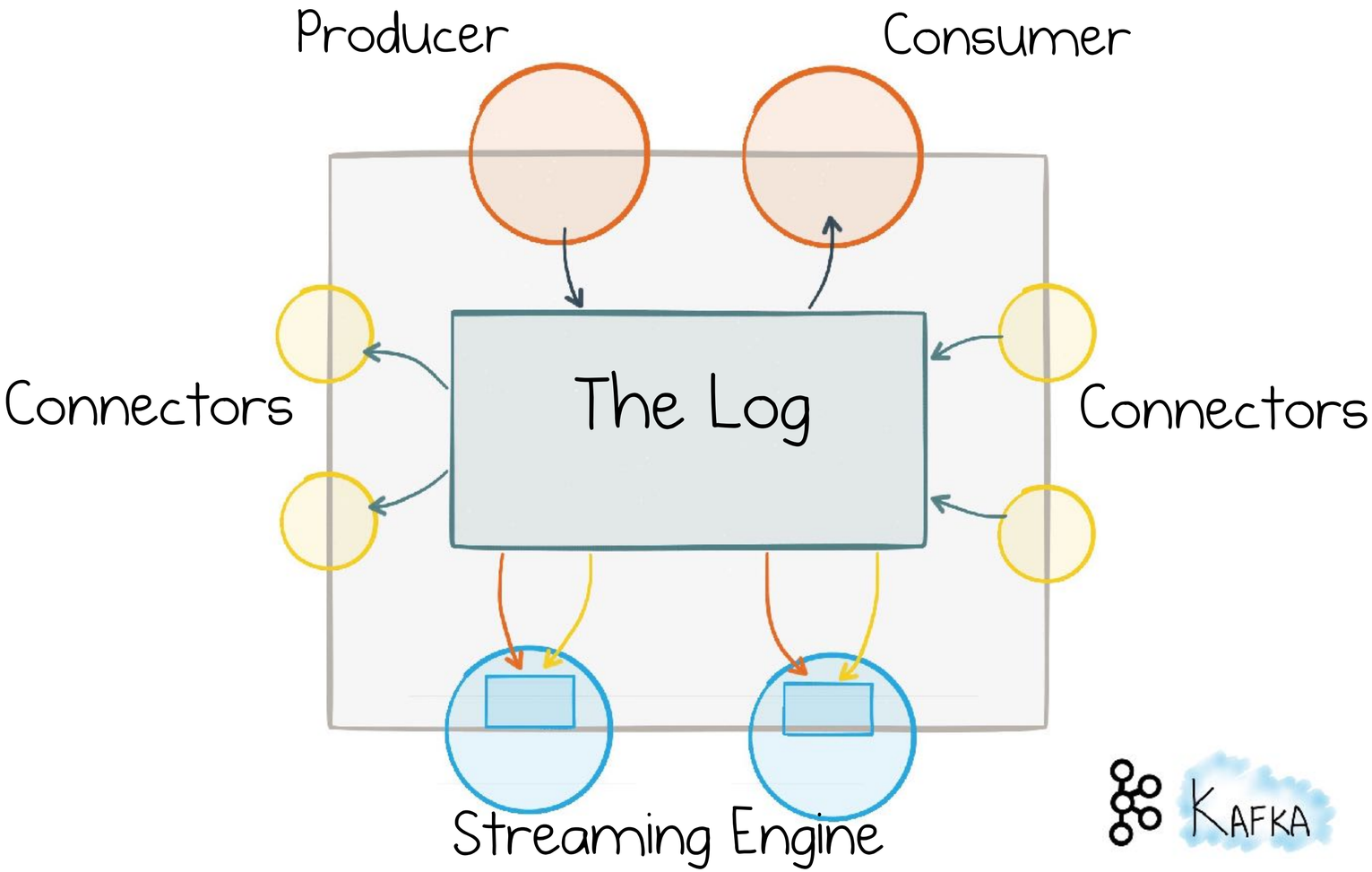
https://cnfl.io/ksql-workshop-prereq

- Make sure you allocate Docker >=8GB memory
  `docker system info | grep Memory`
- Clone the repo
- Pull the git images as instructed in the doc

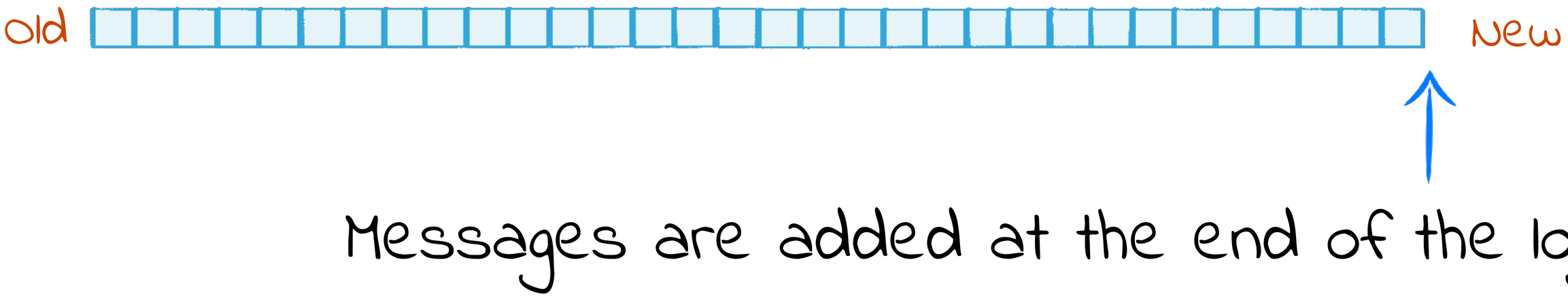
https://cnfl.io/start-ksql-workshop

3. Start Confluent Platform

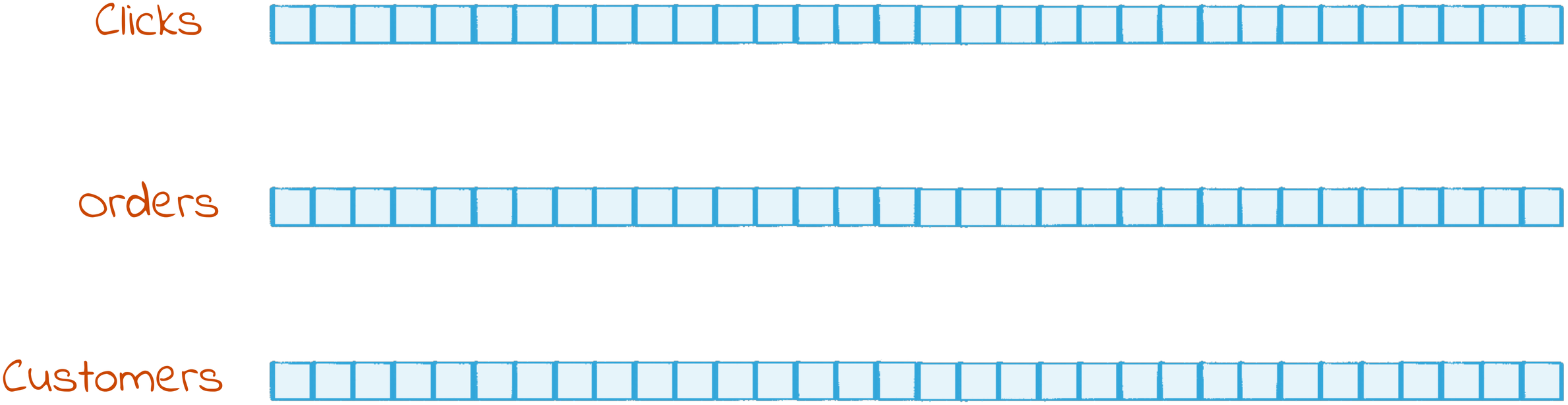
What is an Event Streaming Platform?



Immutable Event Log

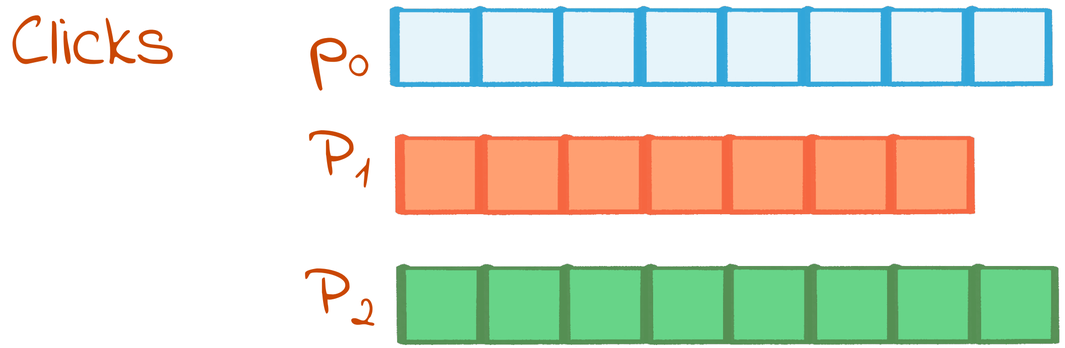


Topics



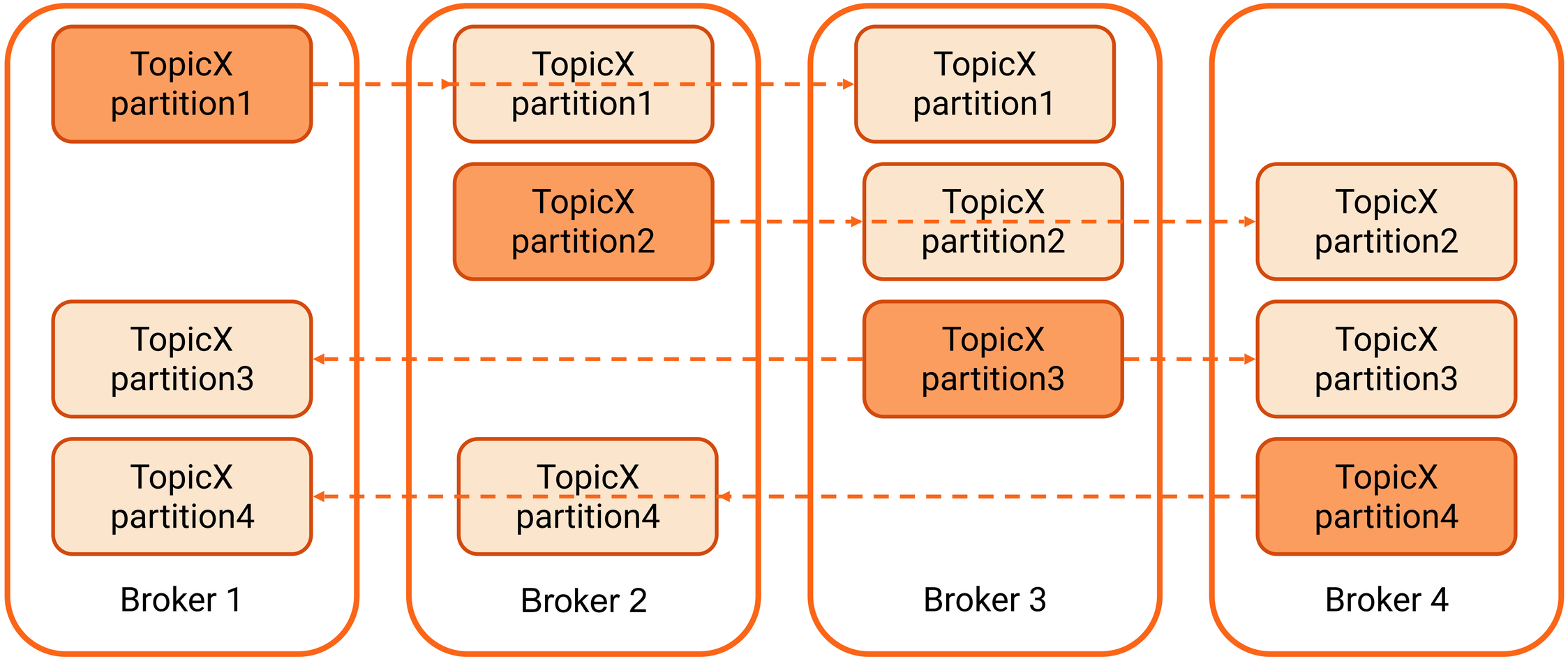
Topics are similar in concept to tables in a database

Partitions

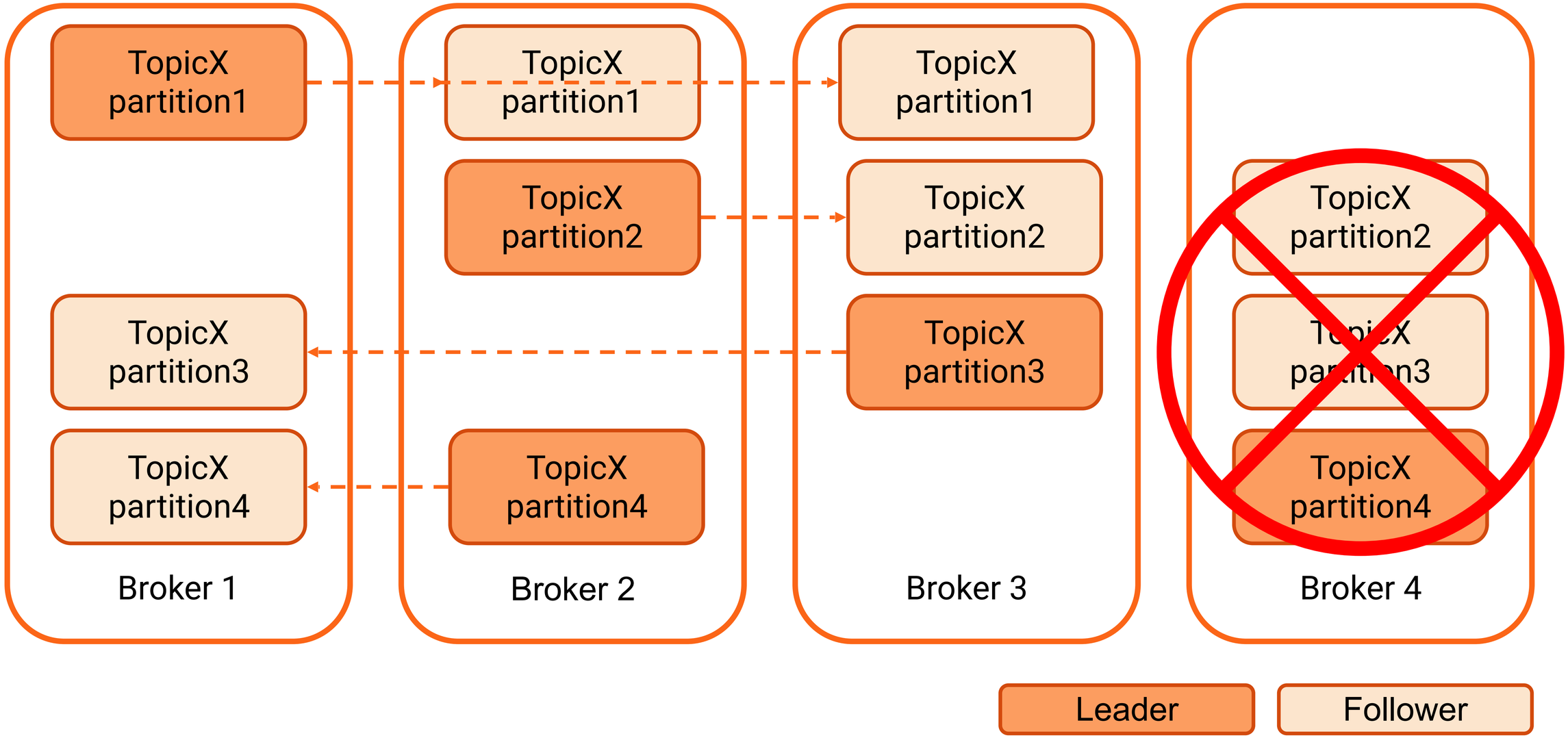


Messages are guaranteed to be strictly ordered within a partition

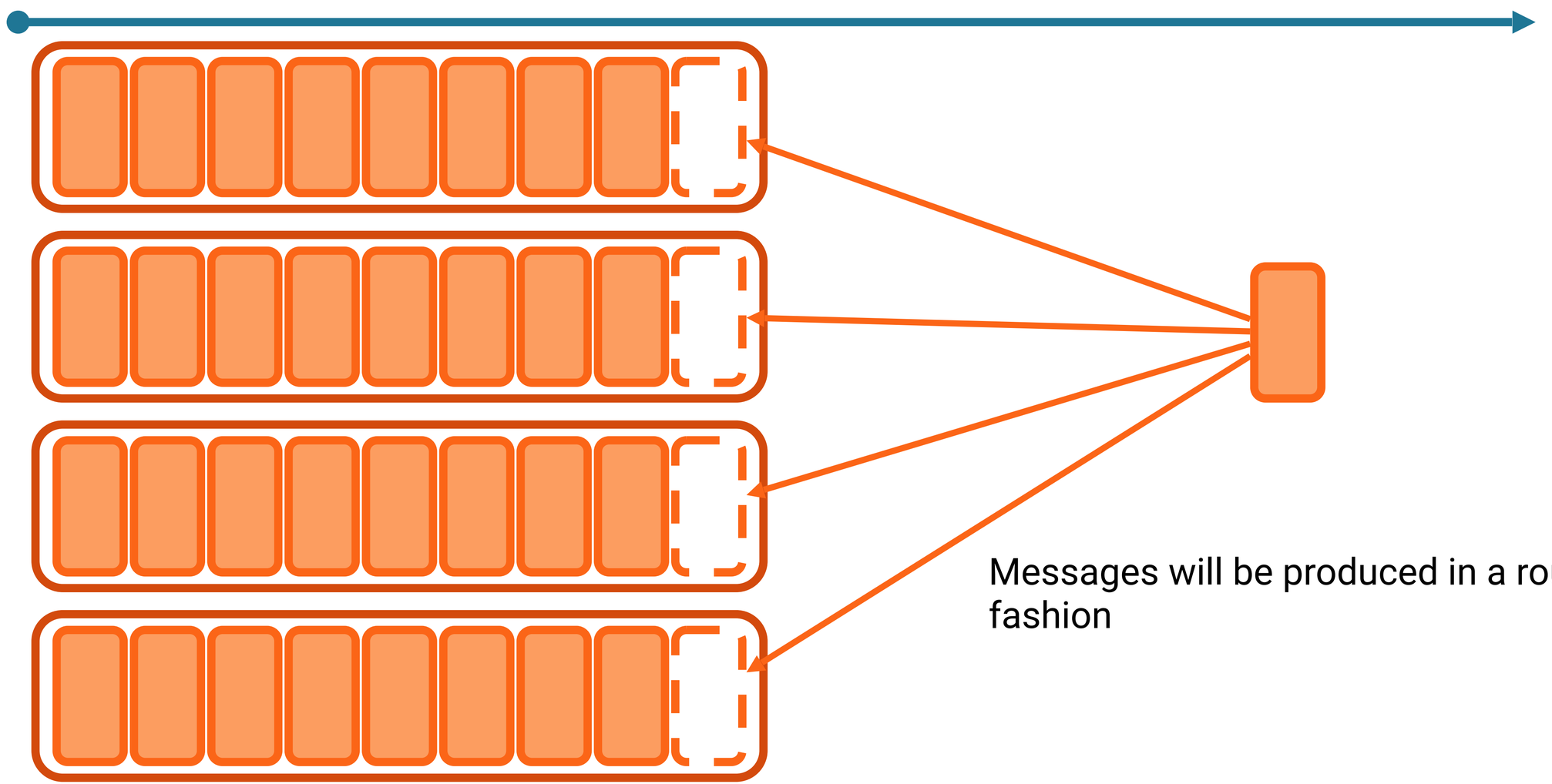
Partition Leadership and Replication



Partition Leadership and Replication - node failure

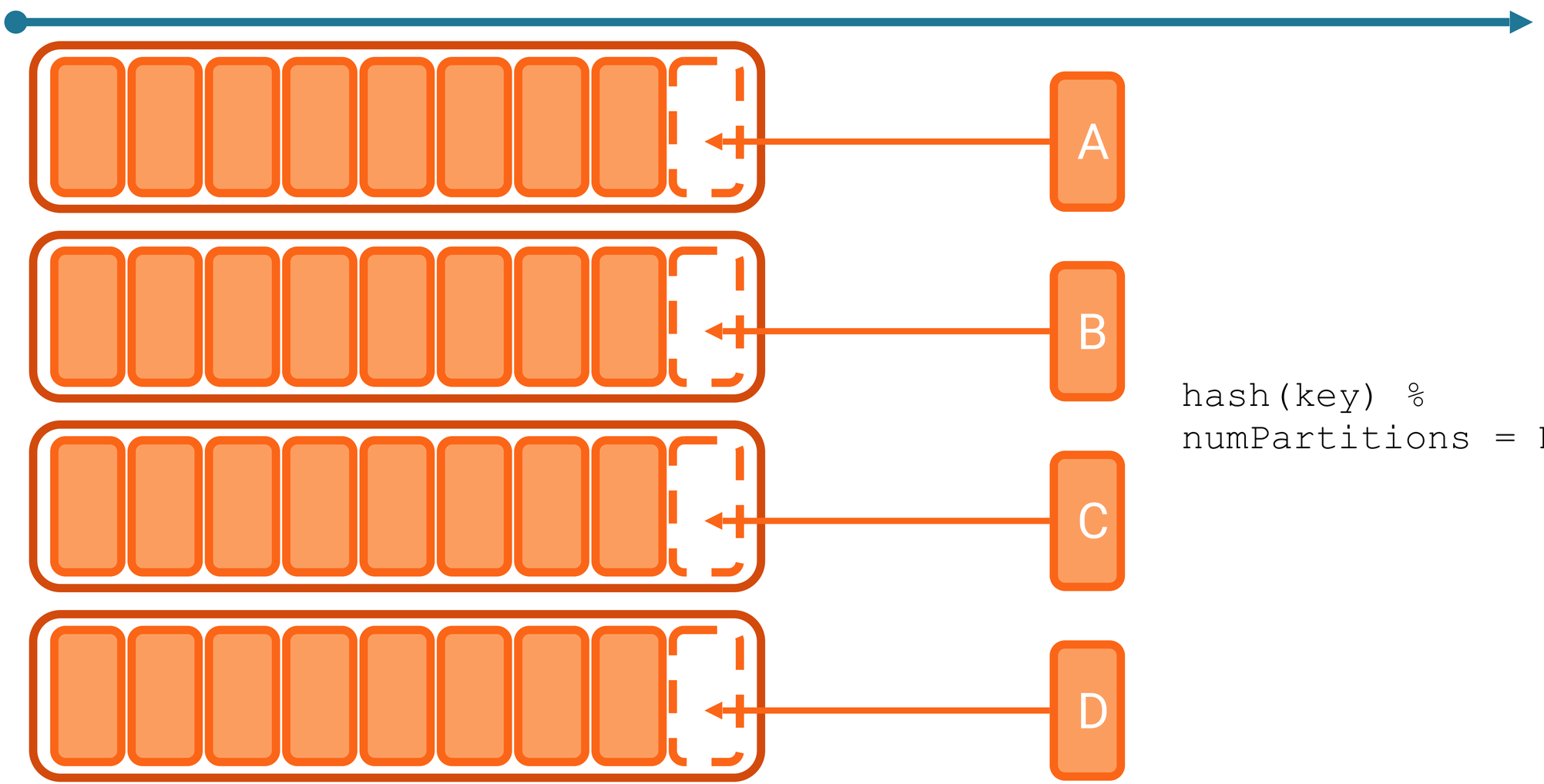


Producing to Kafka - No Key



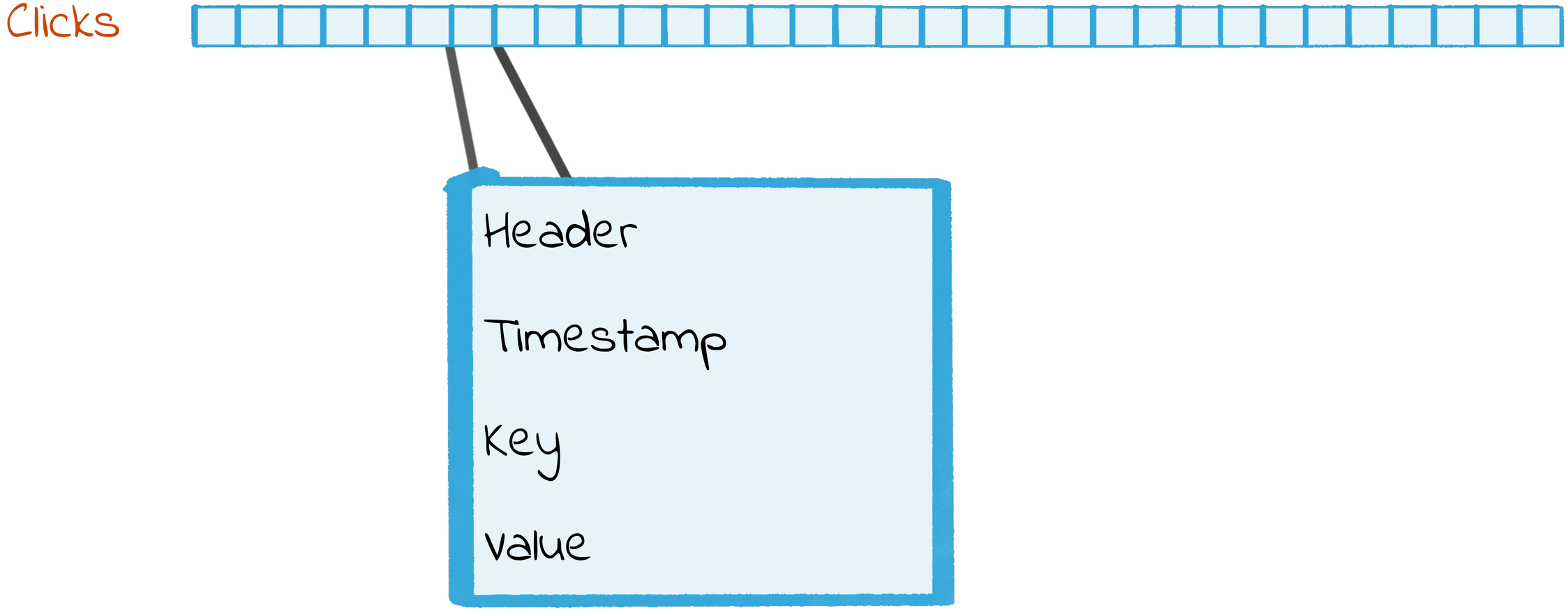
Messages will be produced in a round robin fashion

Producing to Kafka - With Key



Messages are just K/V bytes

plus headers + timestamp



Messages are just K/V bytes

With great power comes great responsibility

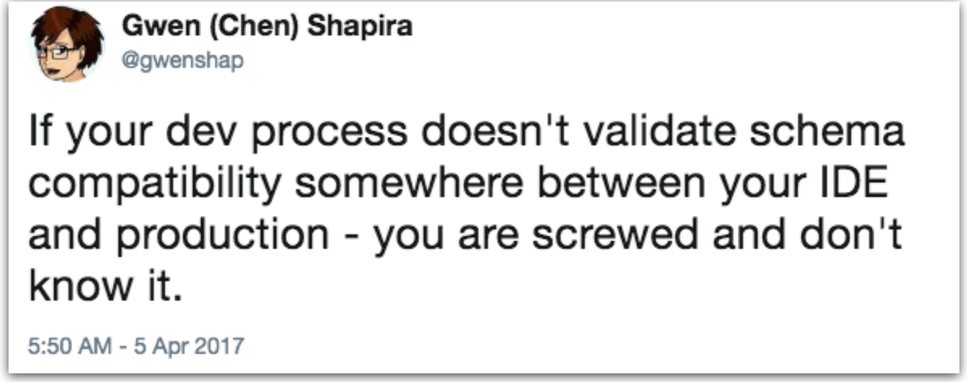
Avro

-> Confluent Schema Registry

Protobuf

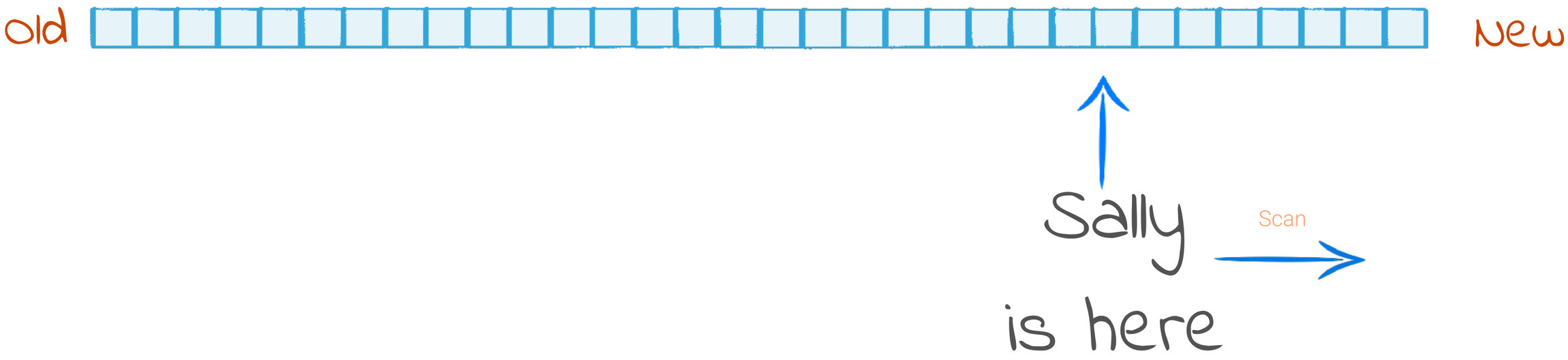
JSON

CSV

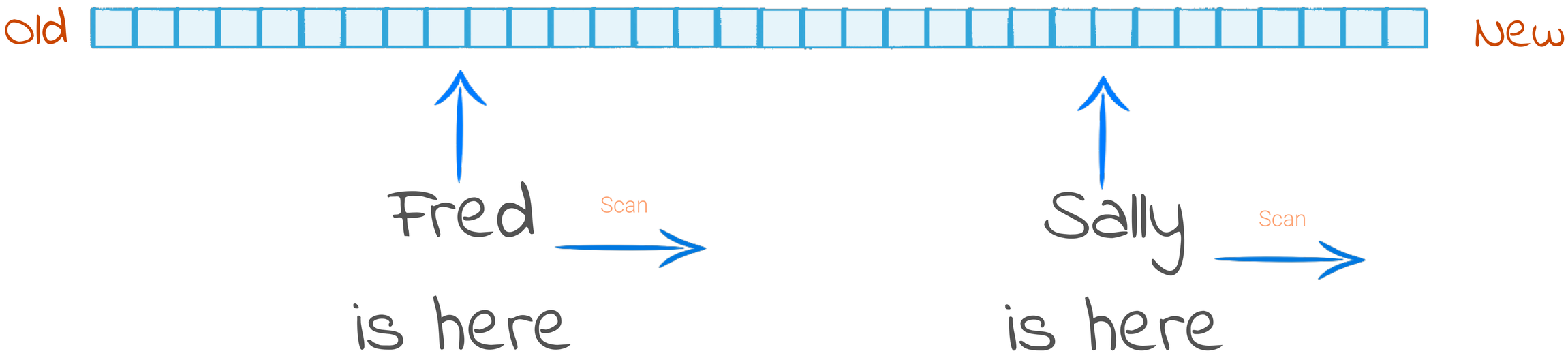


https://qconnewyork.com/system/files/presentation-slides/qcon_17_-_schemas_and_apis.pdf

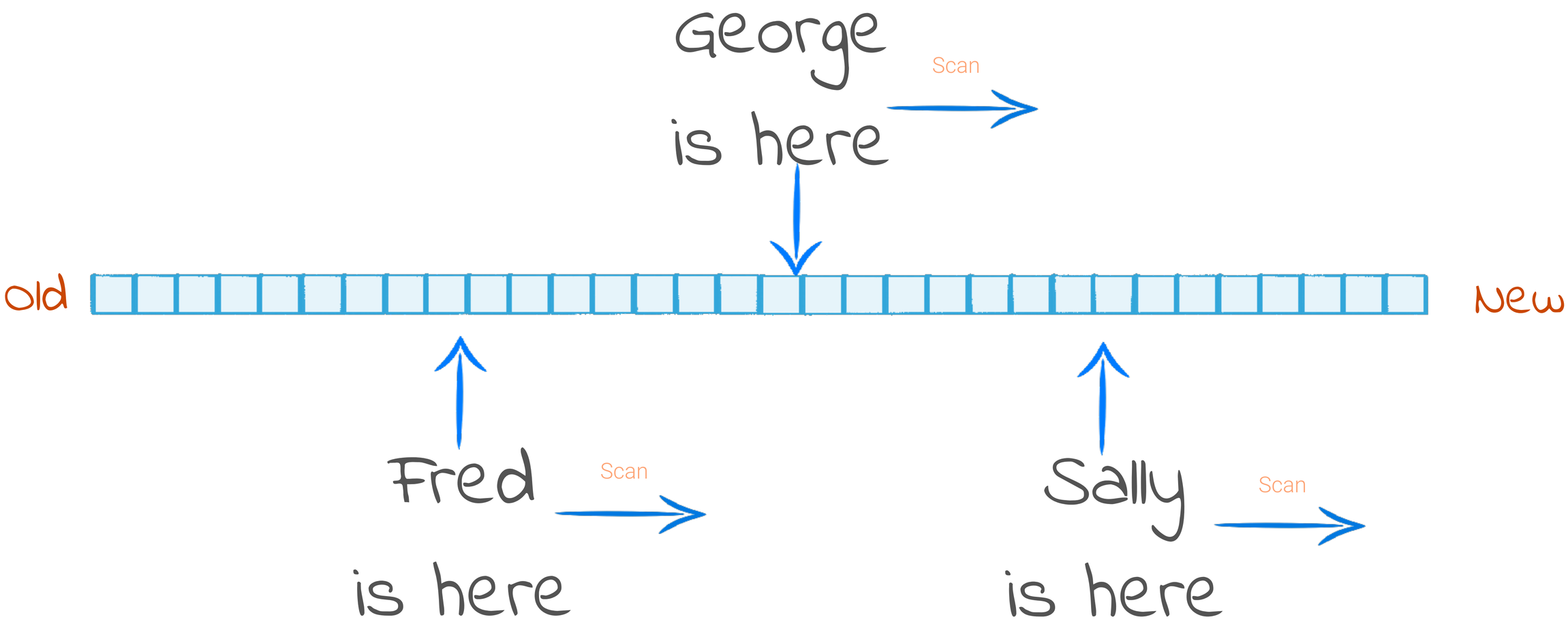
Consumers have a position all of their own



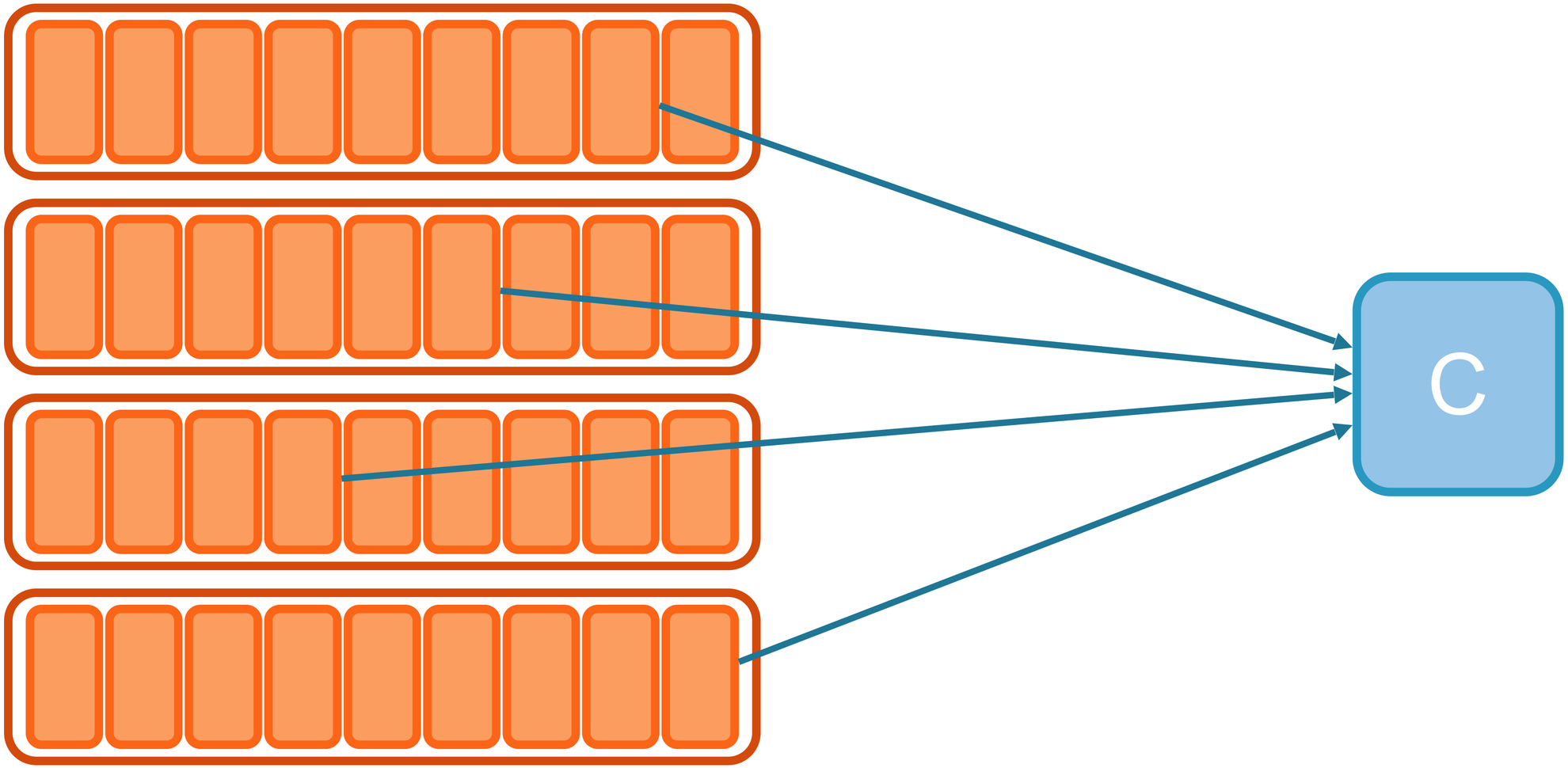
Consumers have a position all of their own



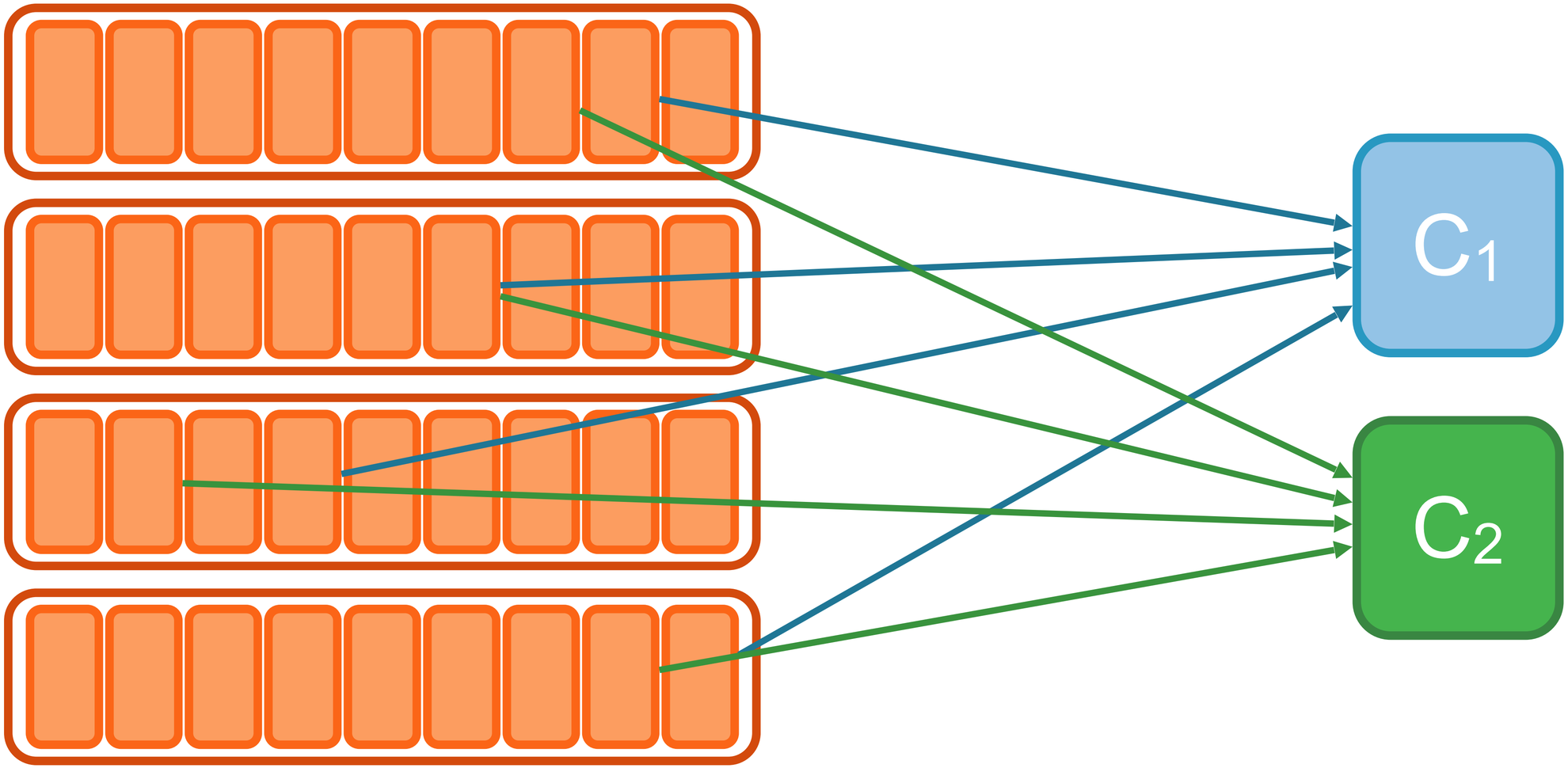
Consumers have a position all of their own



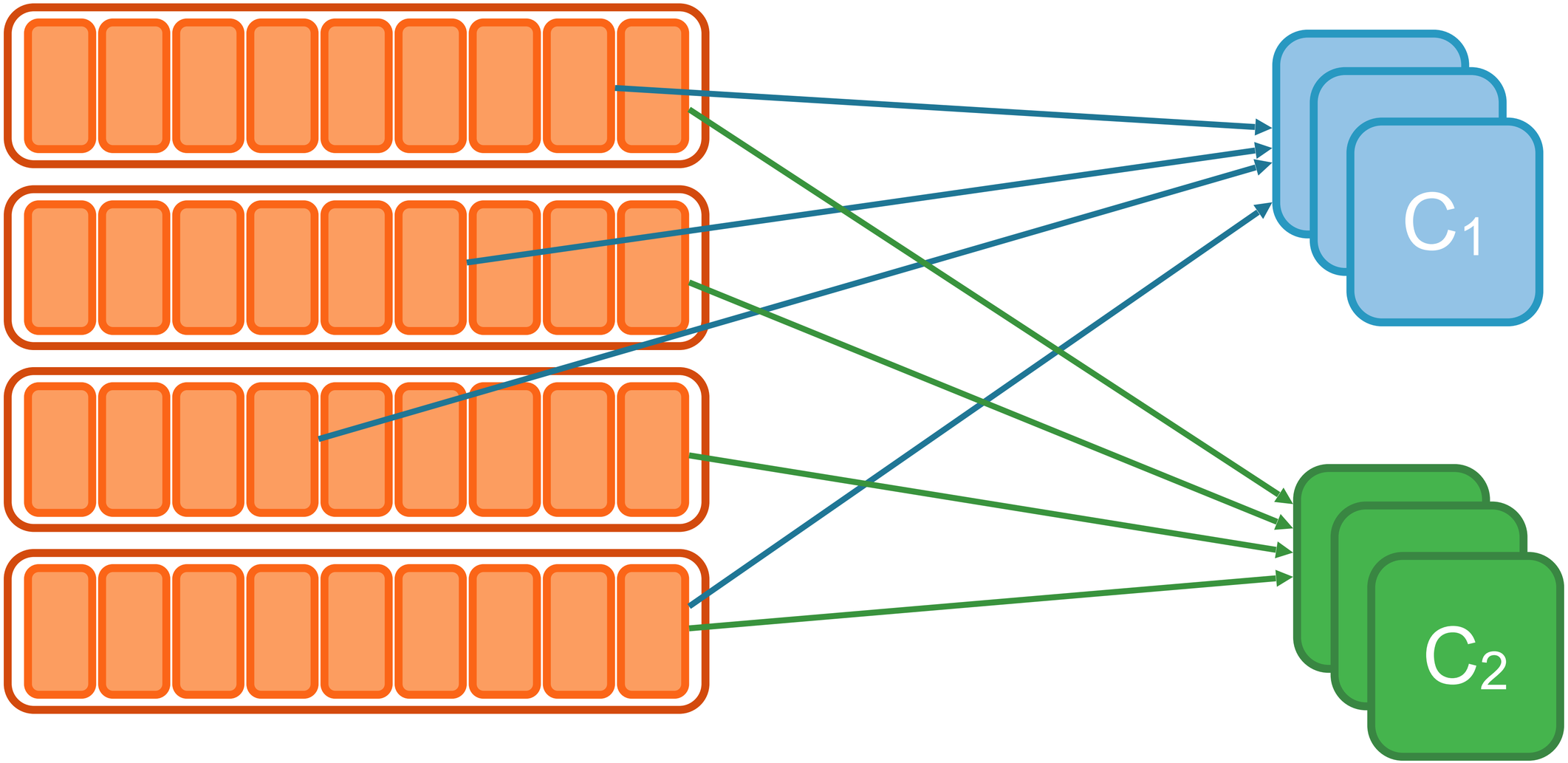
Consuming From Kafka - Single Consumer



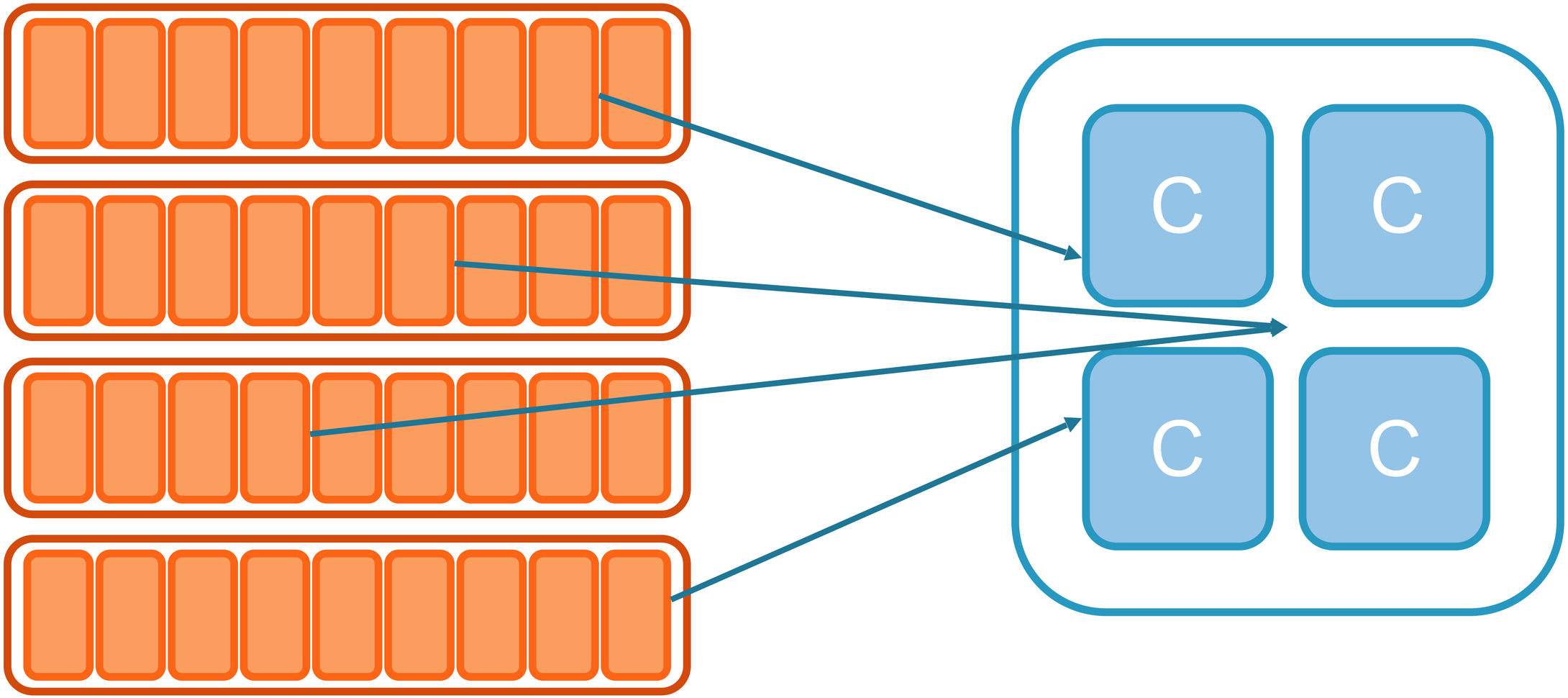
Consuming From Kafka - Multiple Consumers



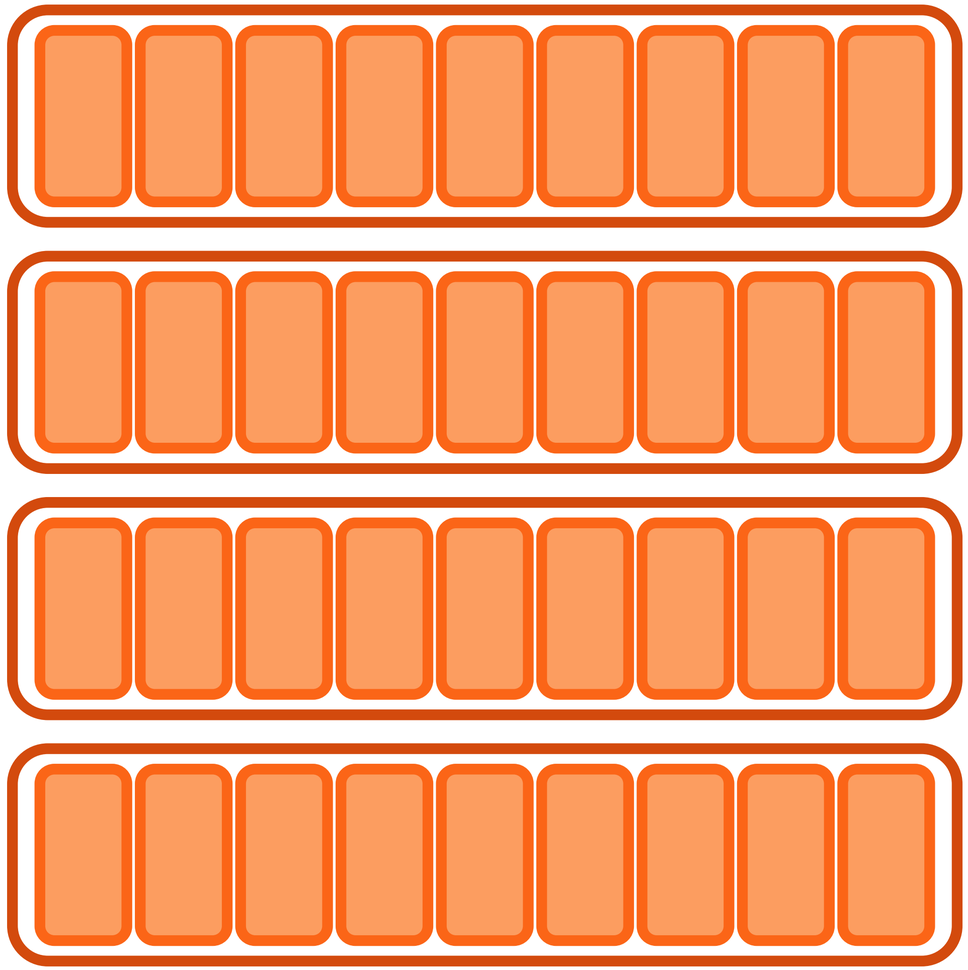
Consuming From Kafka - Grouped Consumers

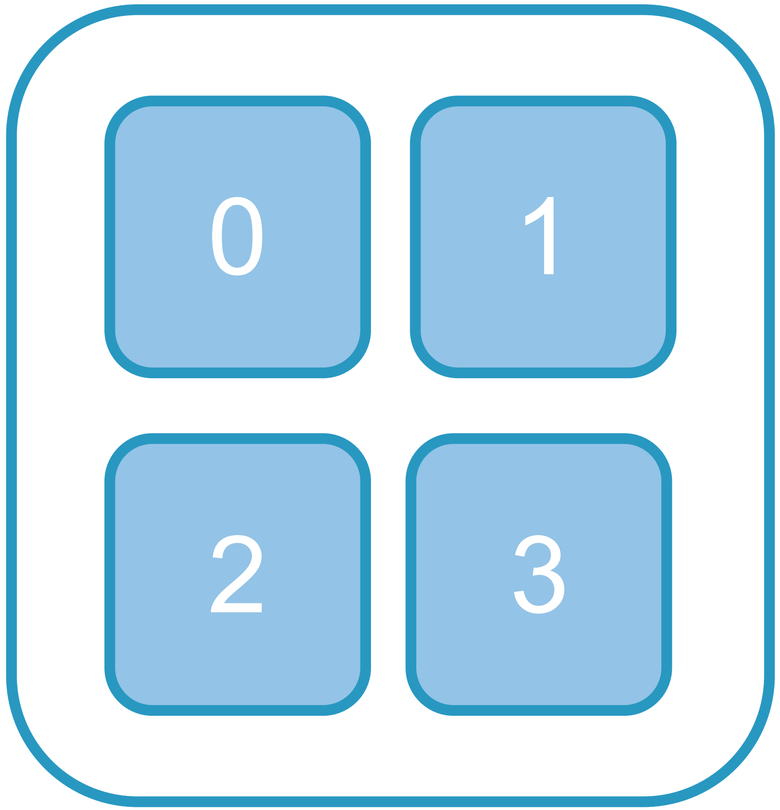


Consuming From Kafka - Grouped Consumers

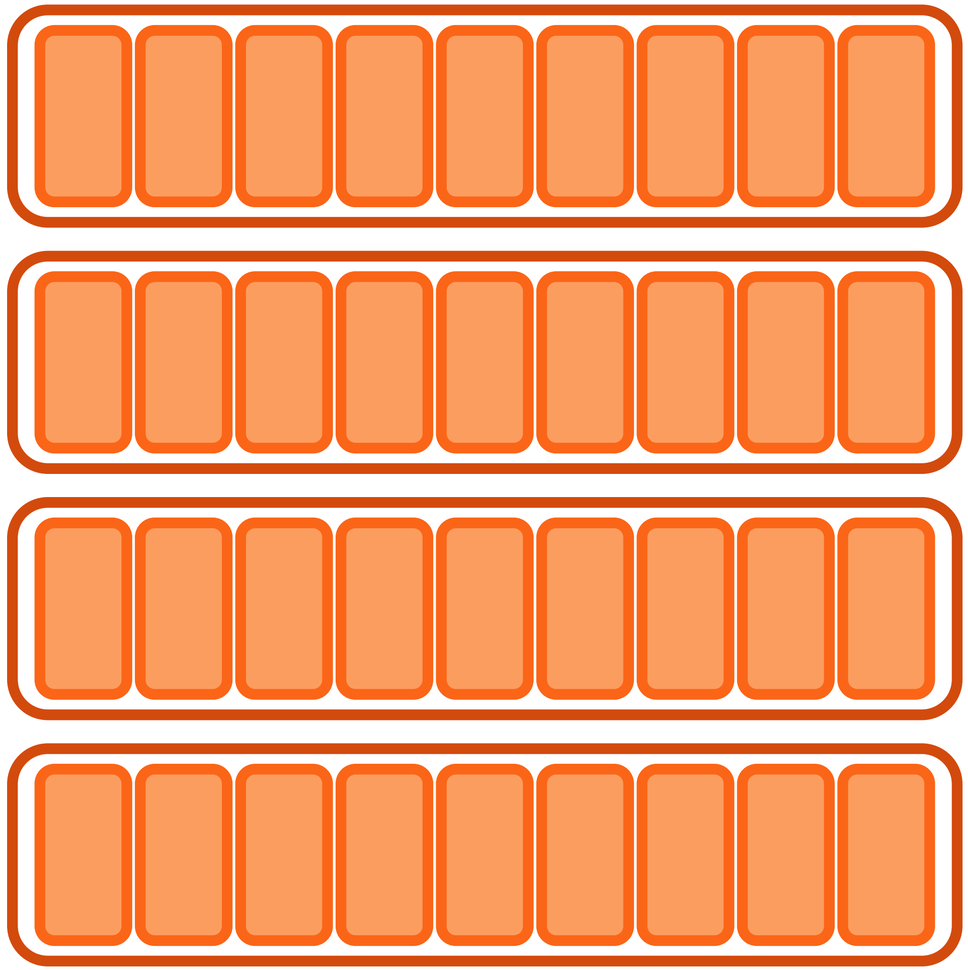


Consuming From Kafka - Grouped Consumers

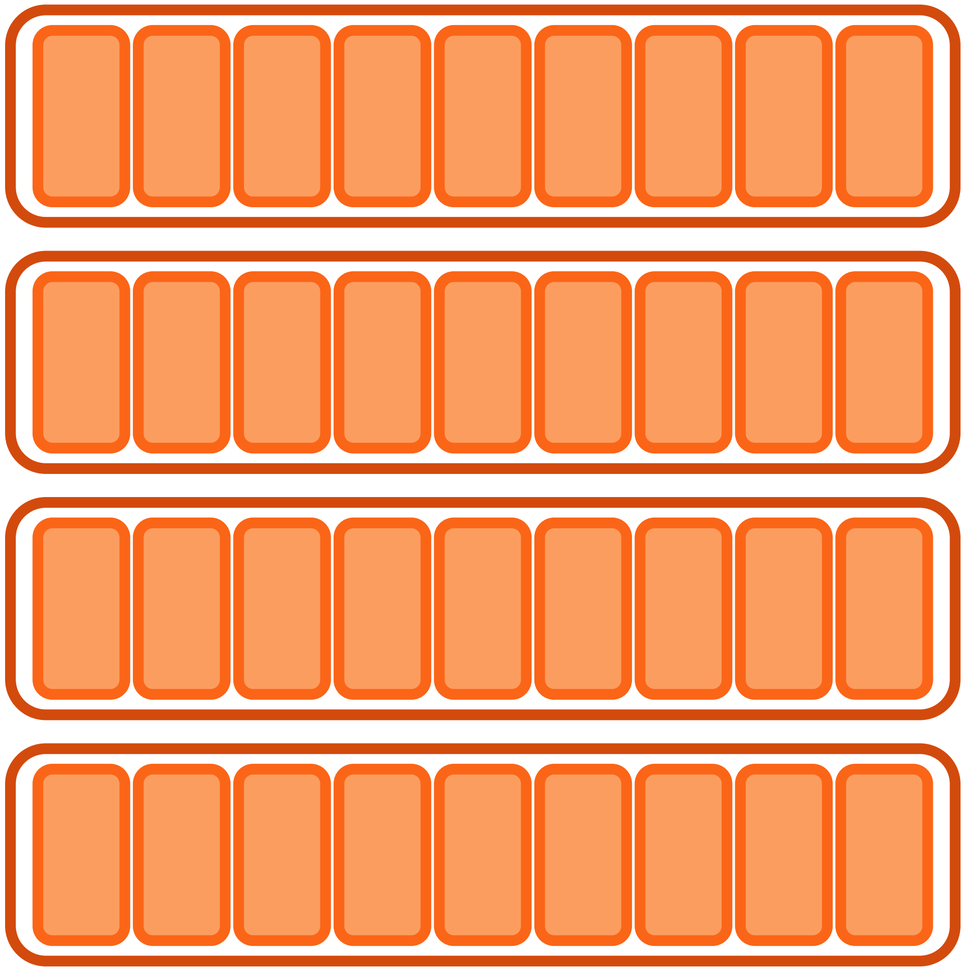


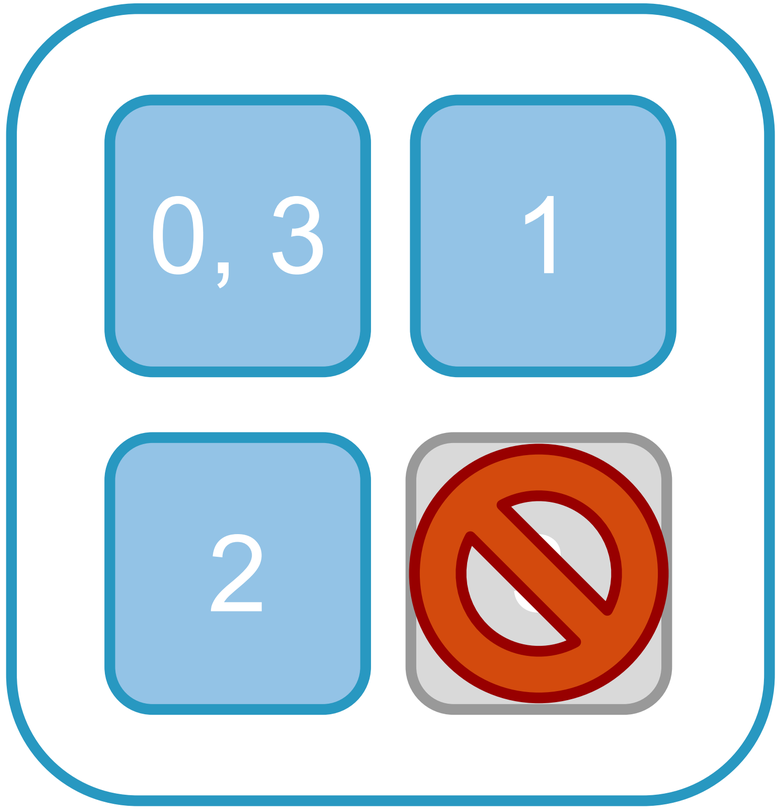


Consuming From Kafka - Grouped Consumers

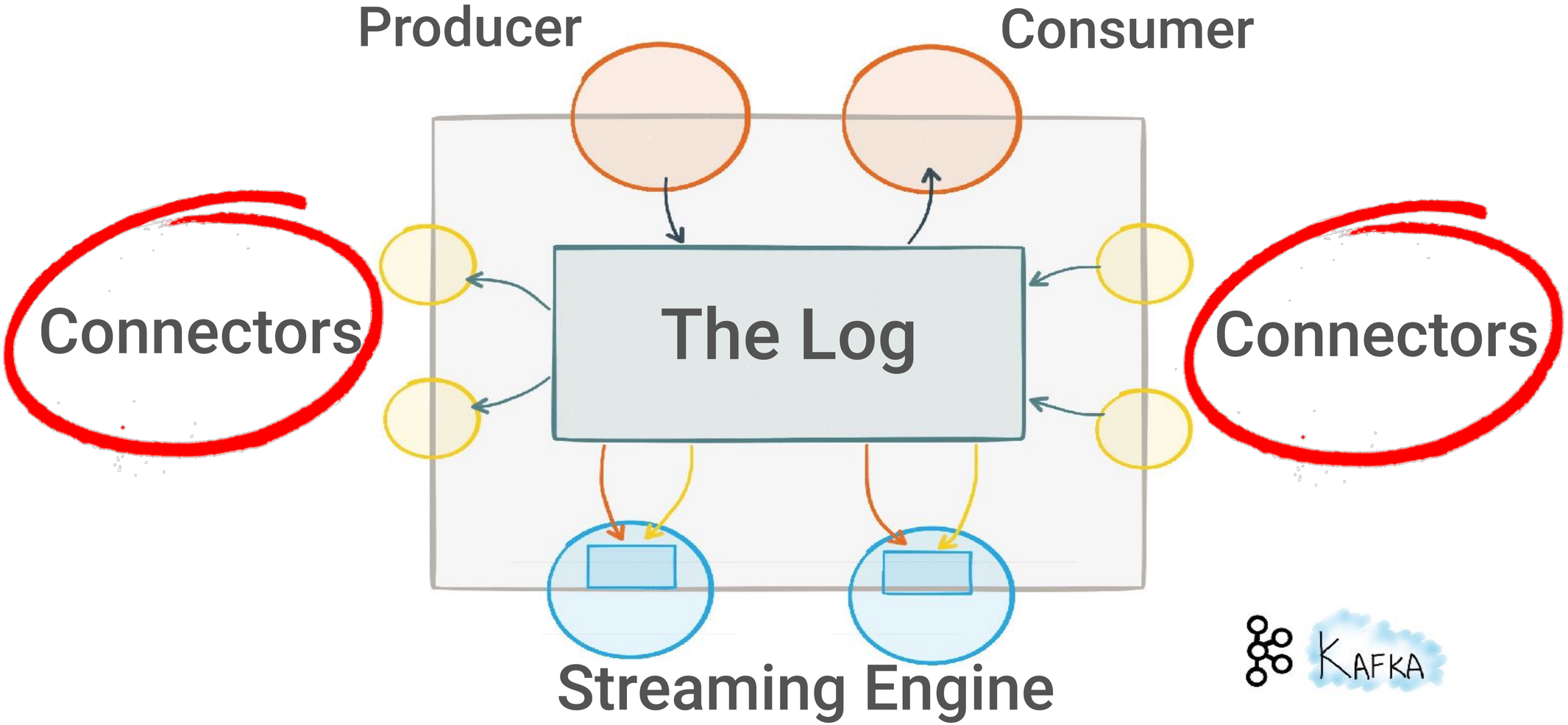


Consuming From Kafka - Grouped Consumers

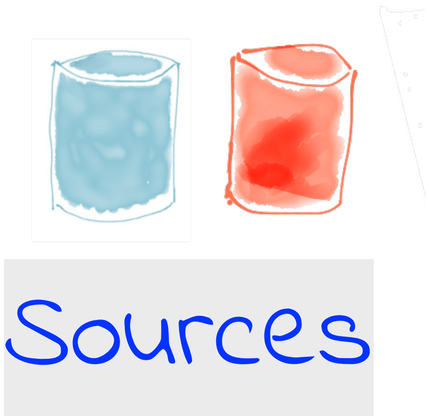




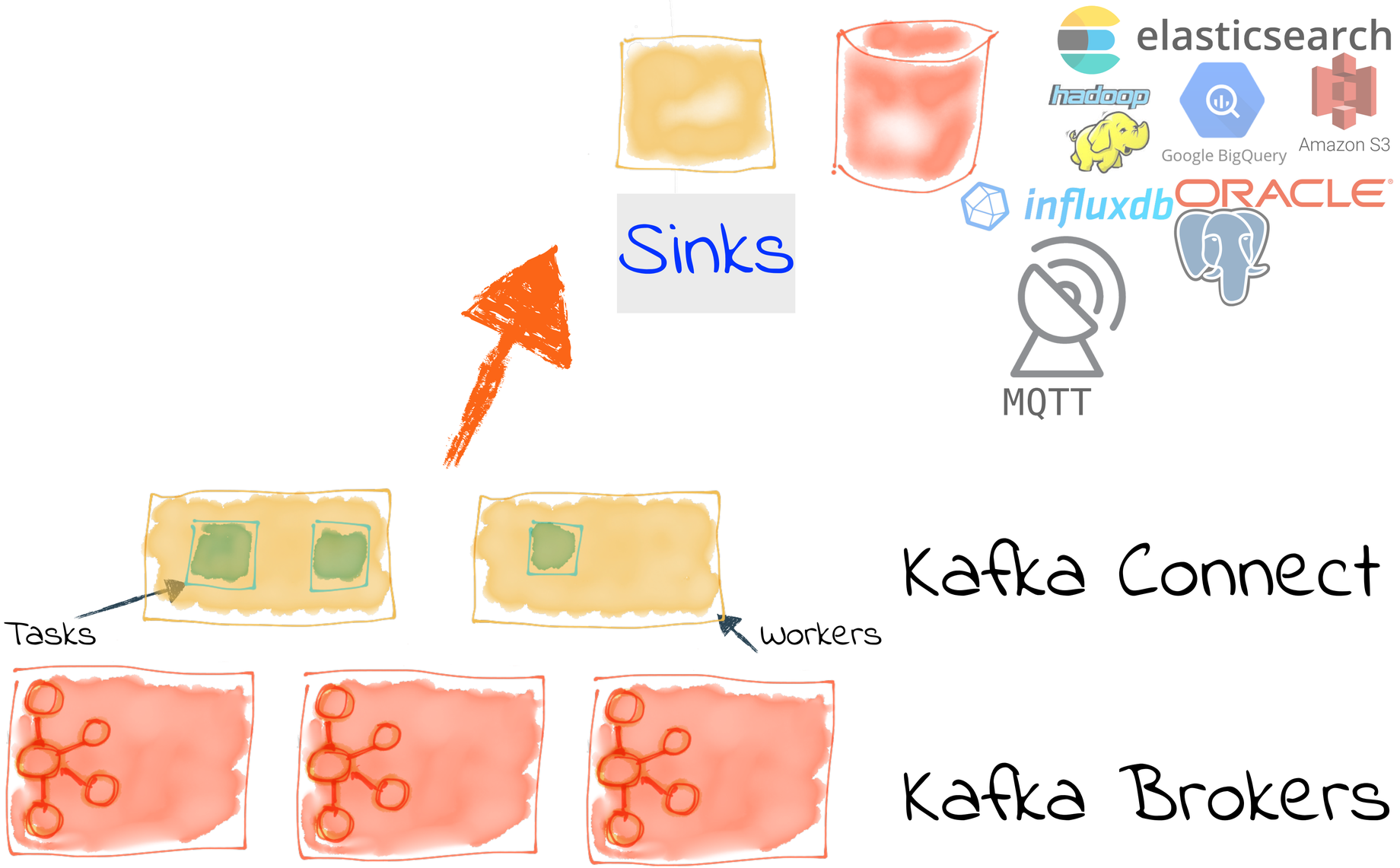
The Connect API



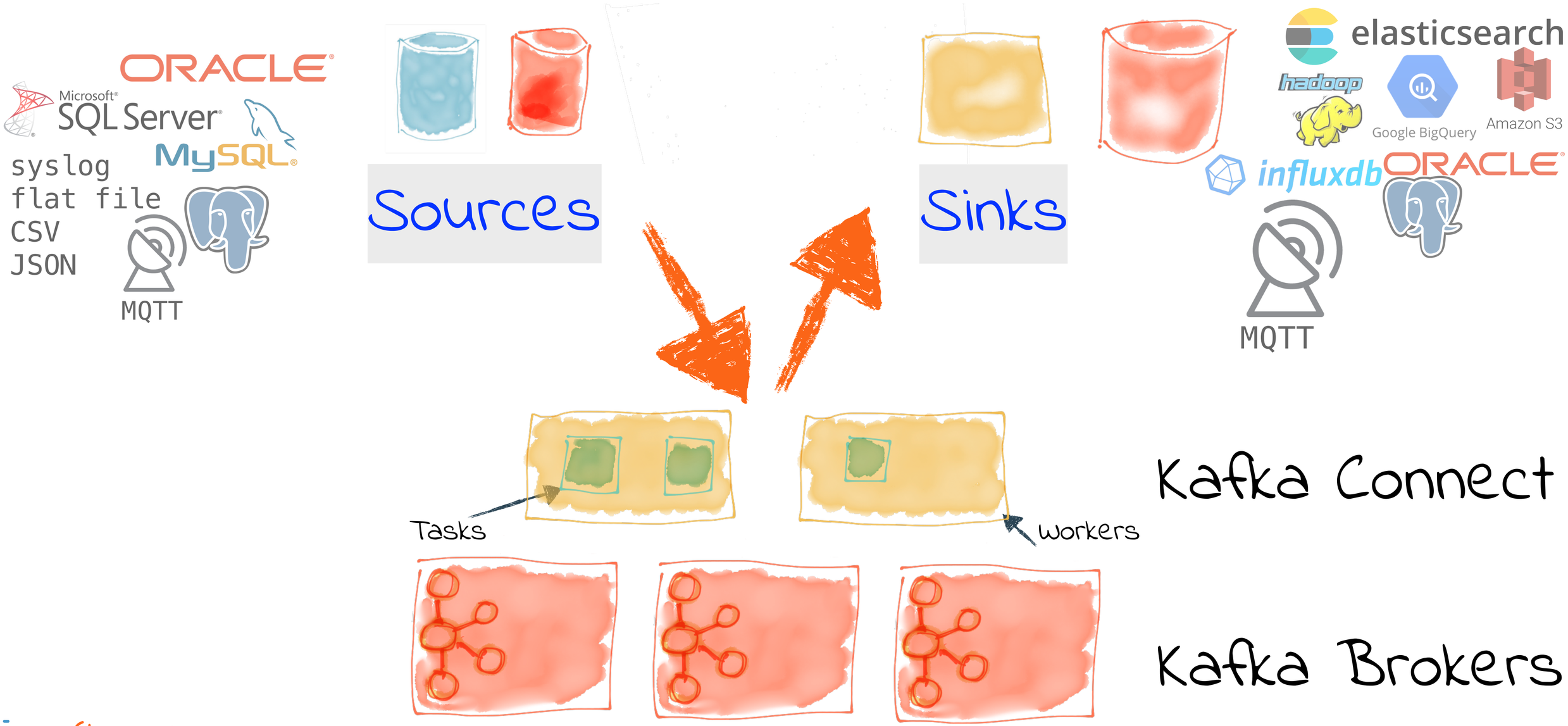
Streaming Integration with Kafka Connect



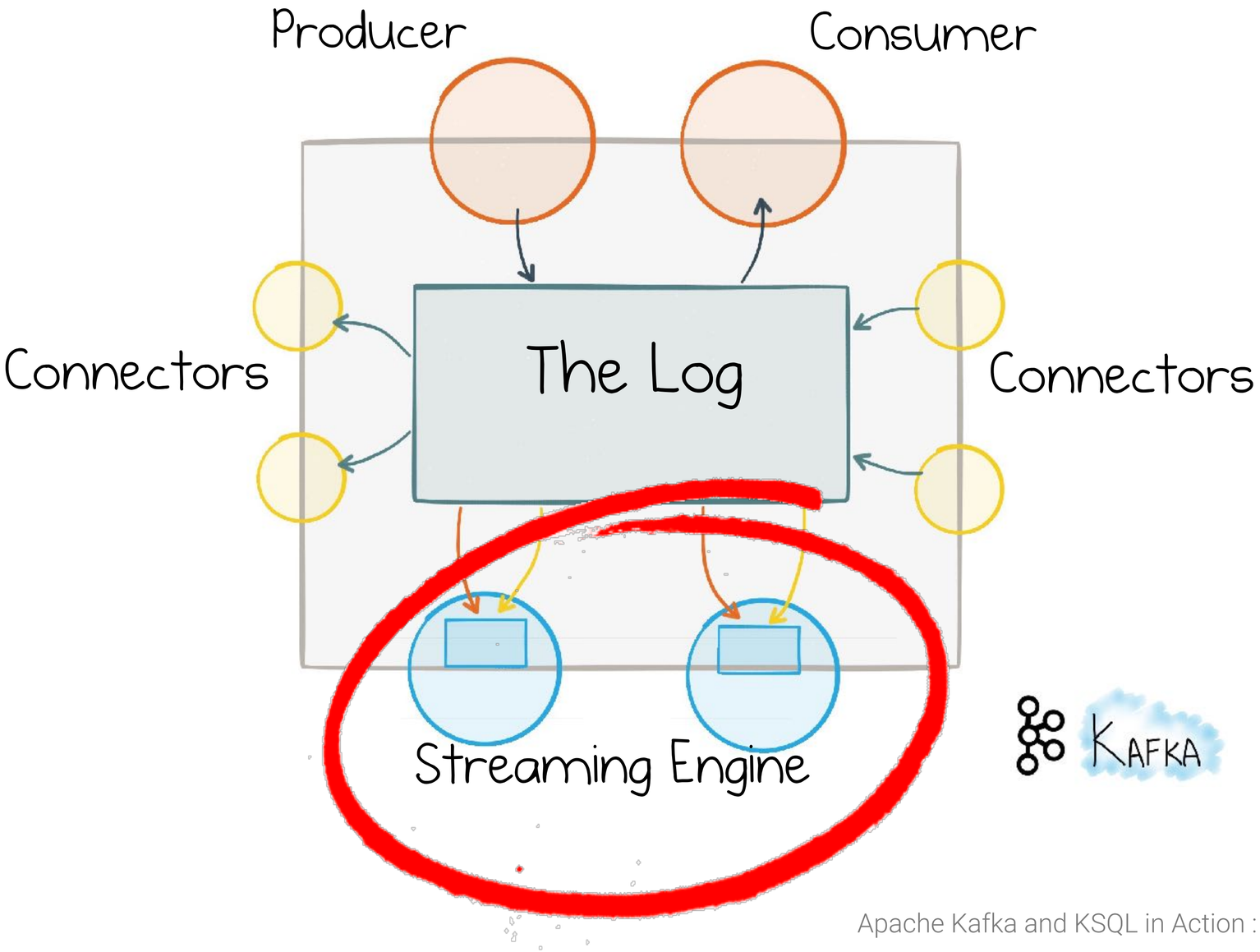
Streaming Integration with Kafka Connect



Streaming Integration with Kafka Connect

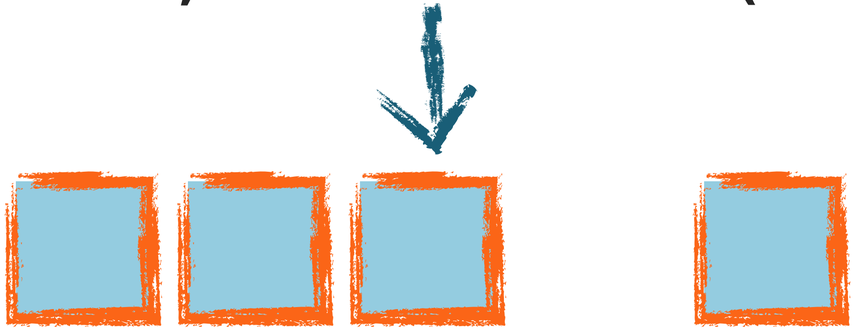


Stream Processing in Kafka



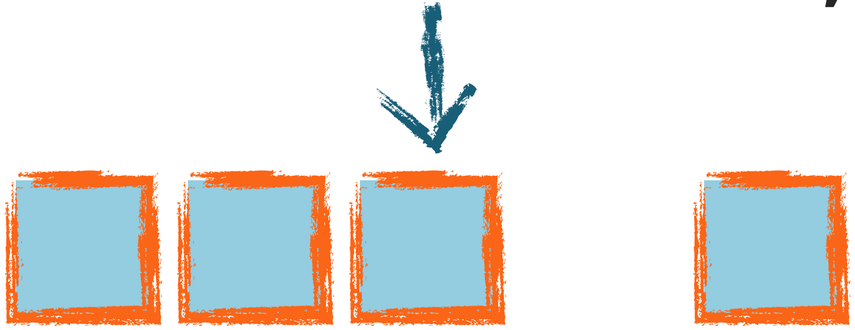
Kafka Streams API

```
final StreamsBuilder builder = new StreamsBuilder()
    .stream("orders", Consumed.with(stringSerde, ordersSerde))
    .filter( (key, order) -> order.getStatus().equals("COMPLETE") )
    .to("complete_orders", Produced.with(stringSerde, ordersSerde));
```

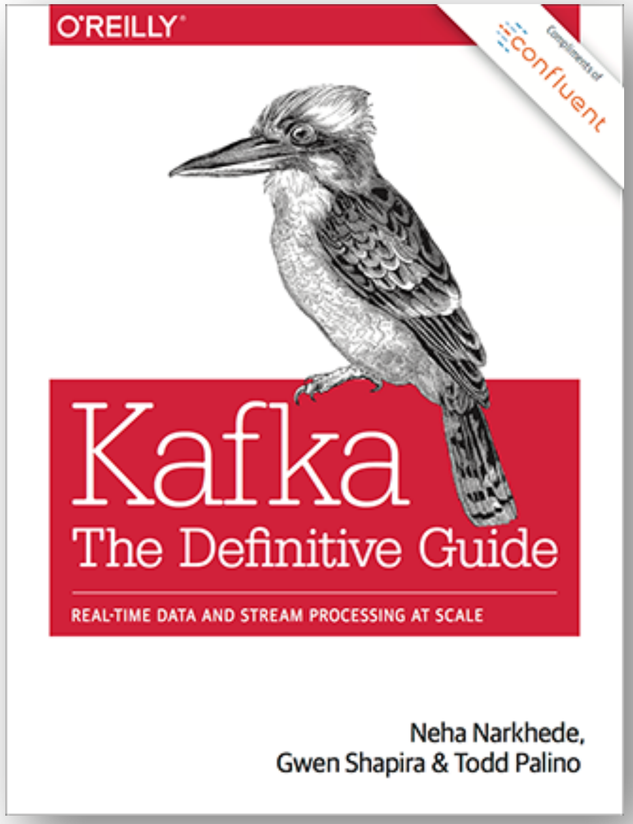


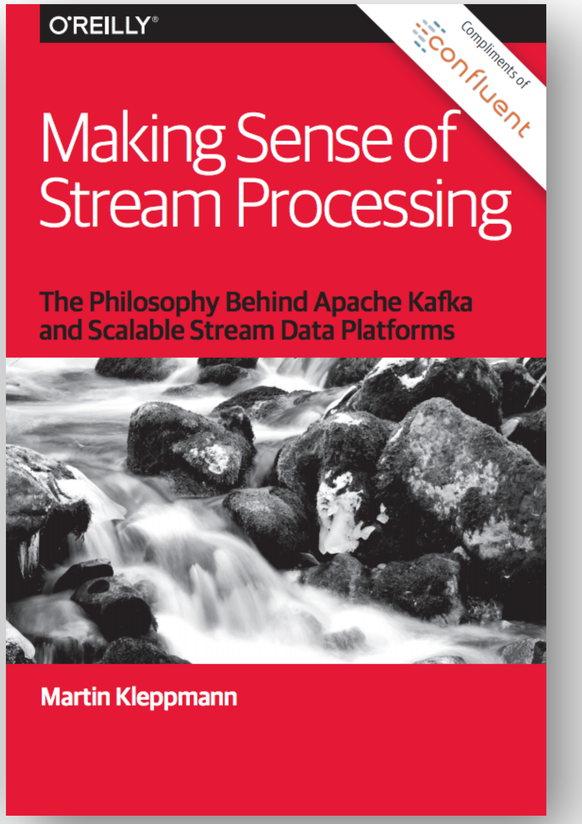
Stream Processing with KSQL

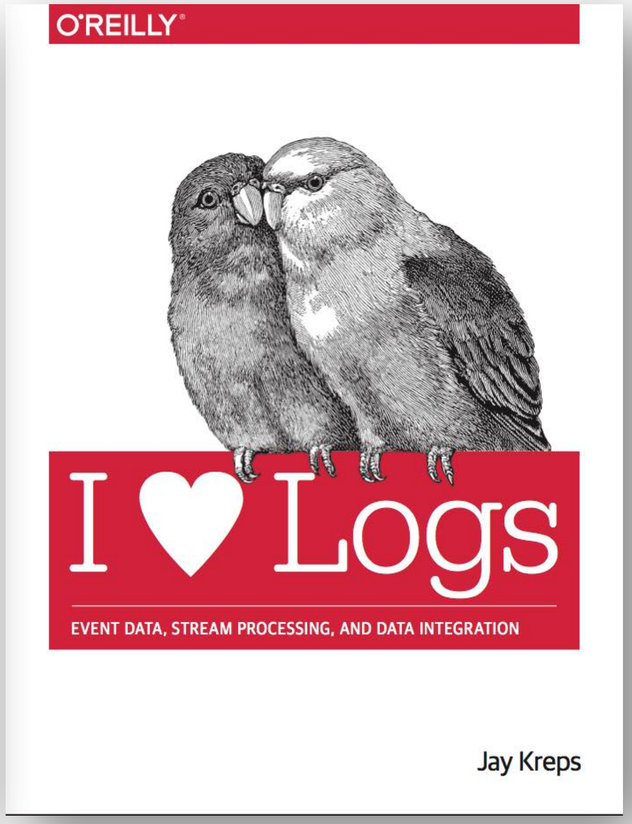
```
CREATE STREAM completedOrders AS
SELECT *
FROM orders
WHERE status='COMPLETE';
```

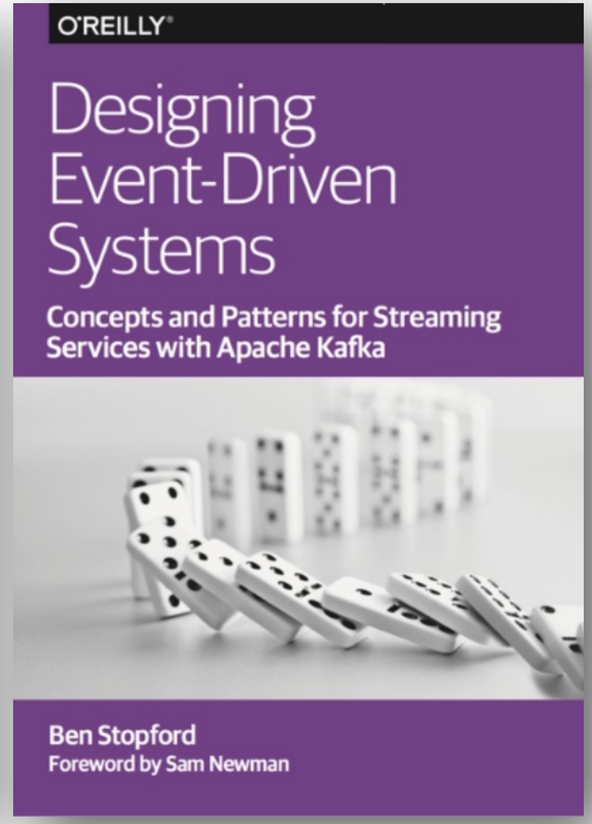


http://cnfl.io/book-bundle

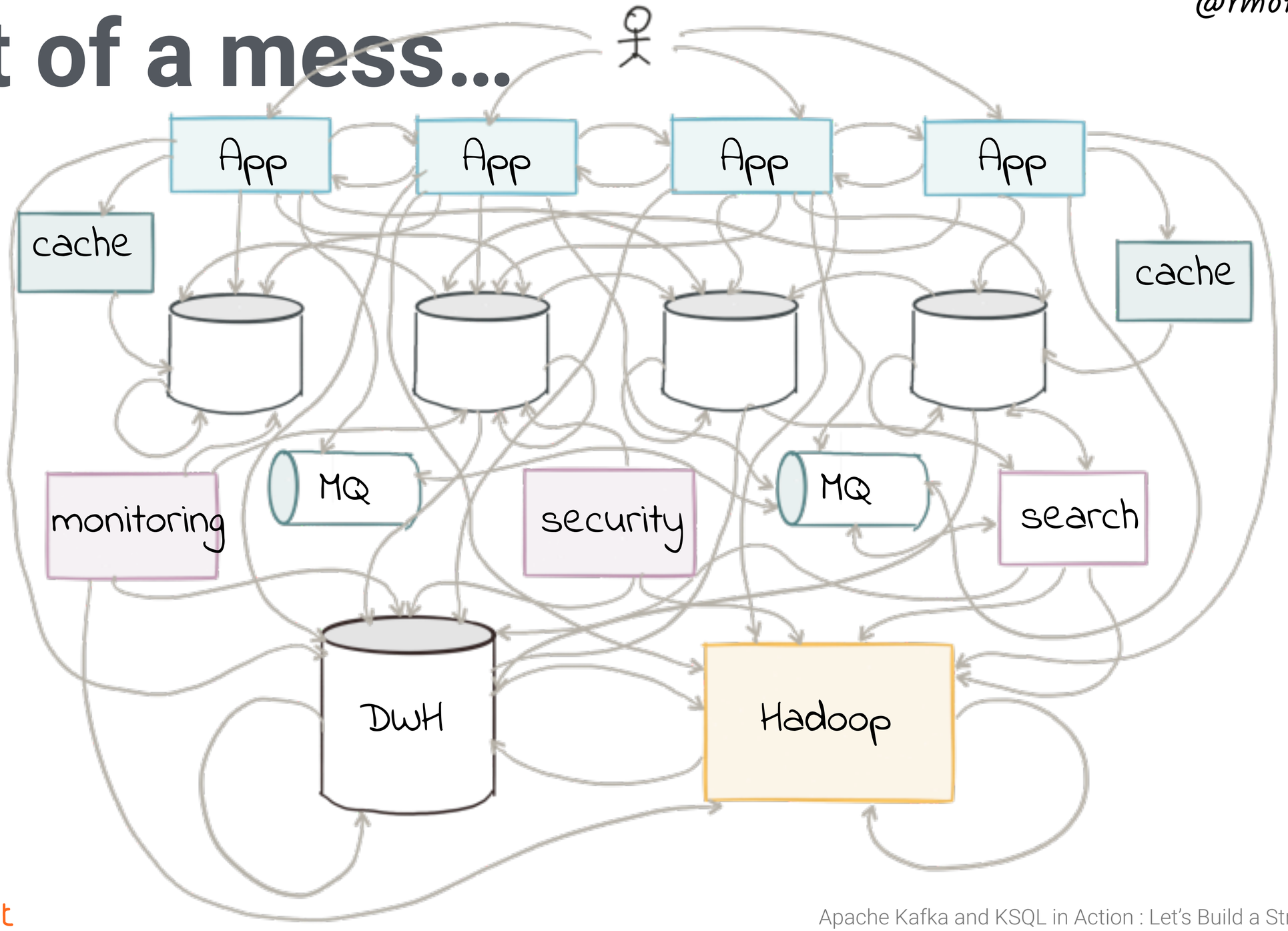




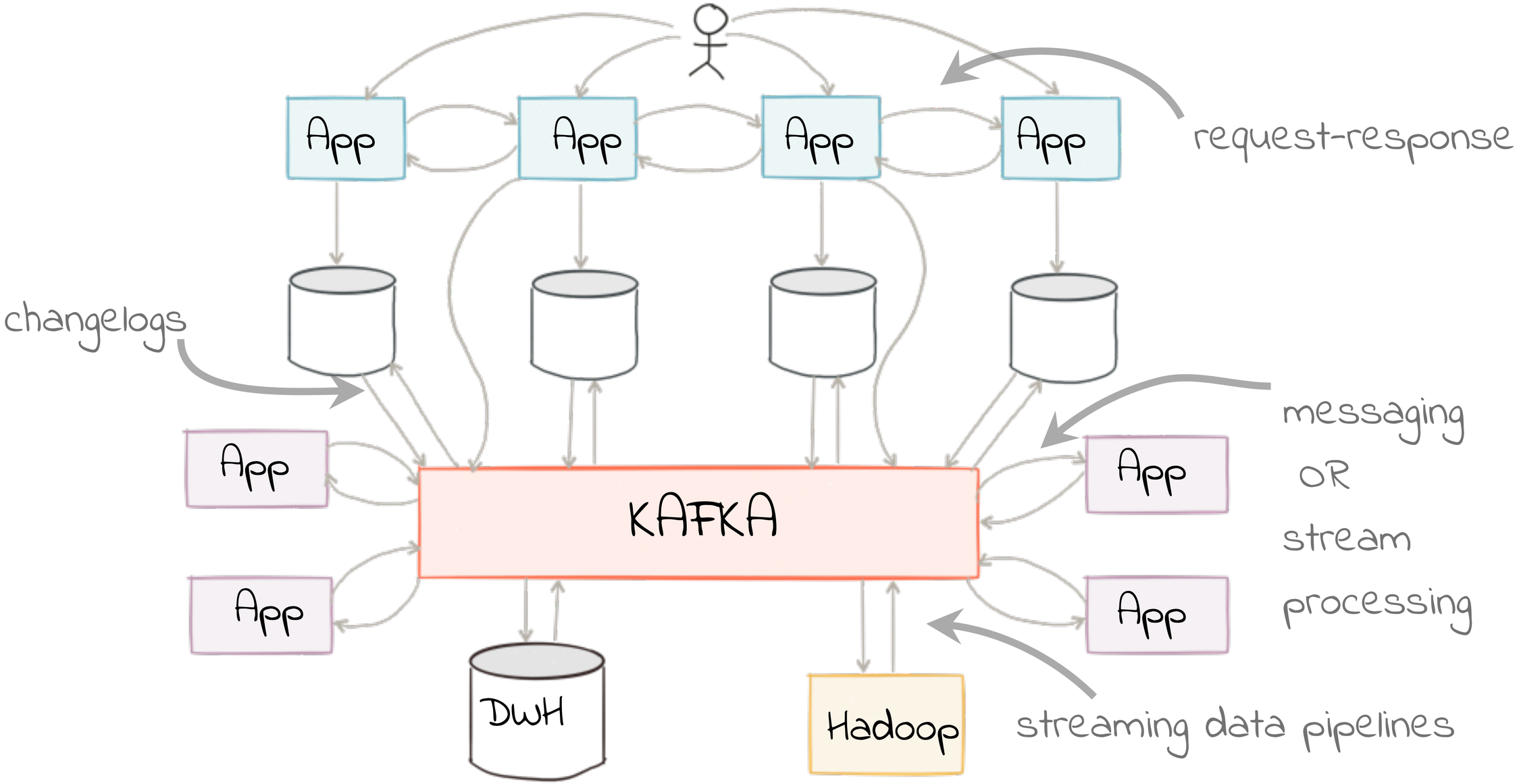




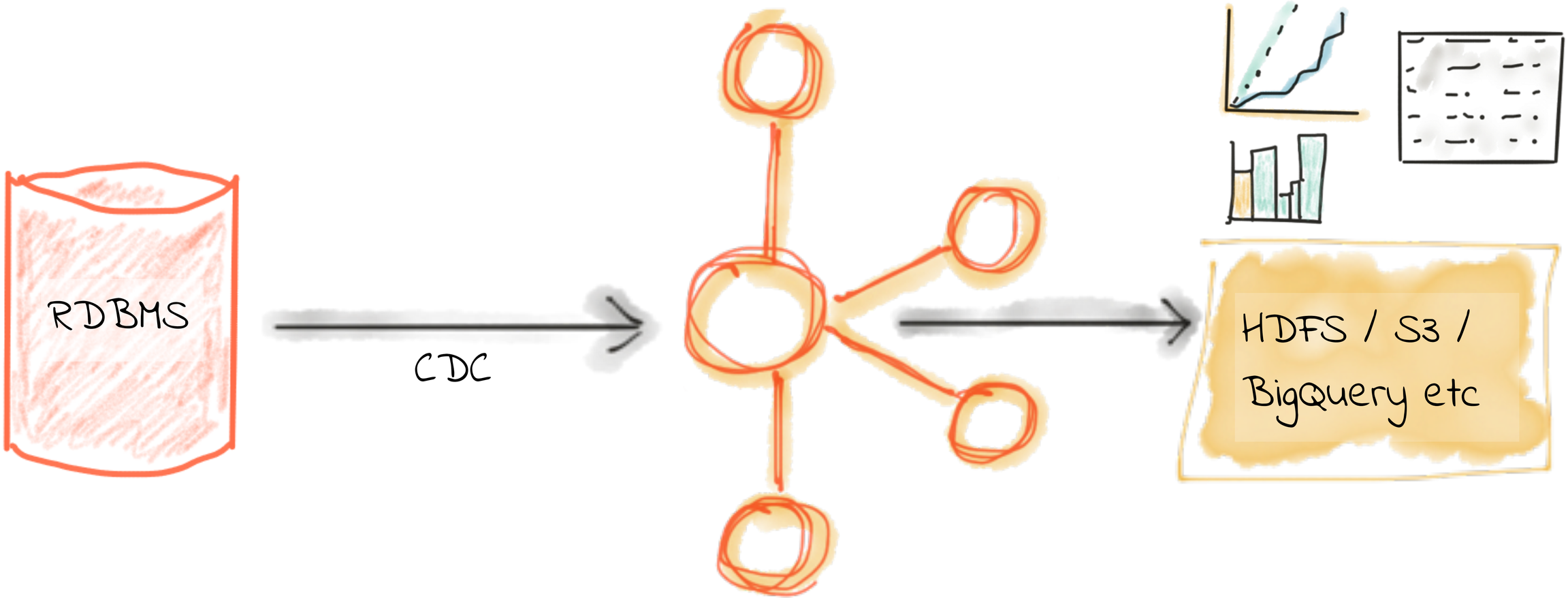
A bit of a mess…



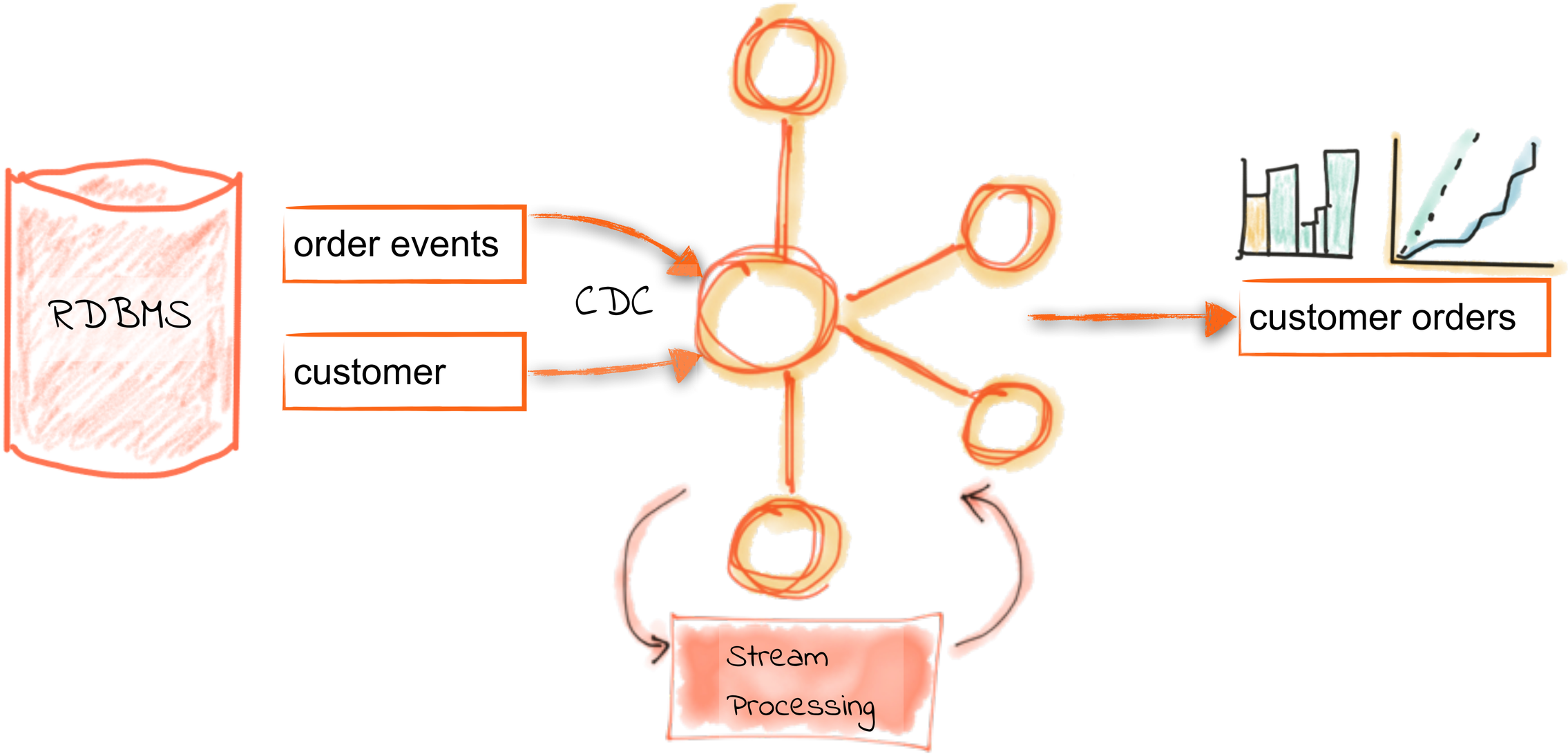
Kafka is a Streaming Platform



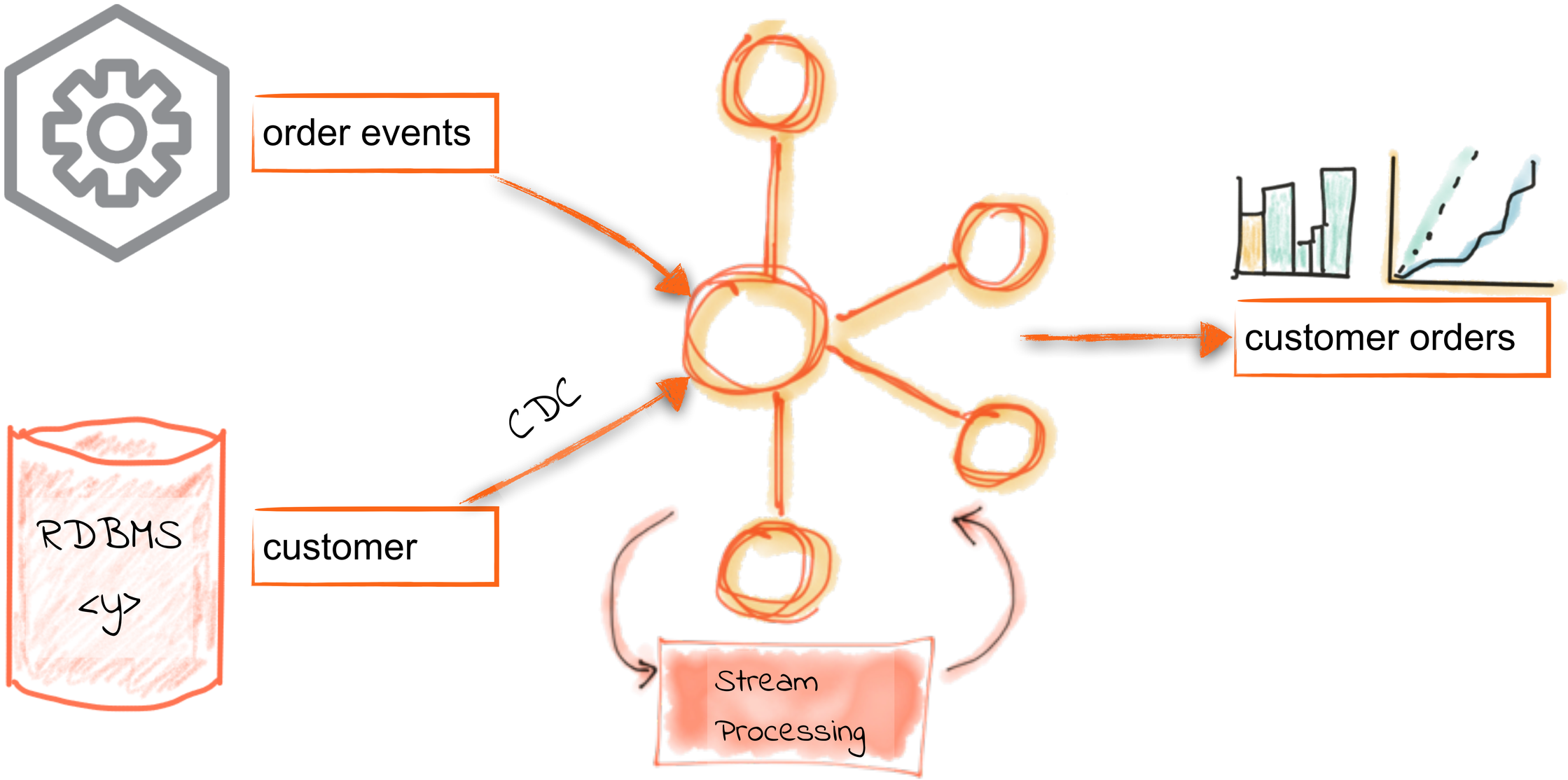
Analytics - Database Offload



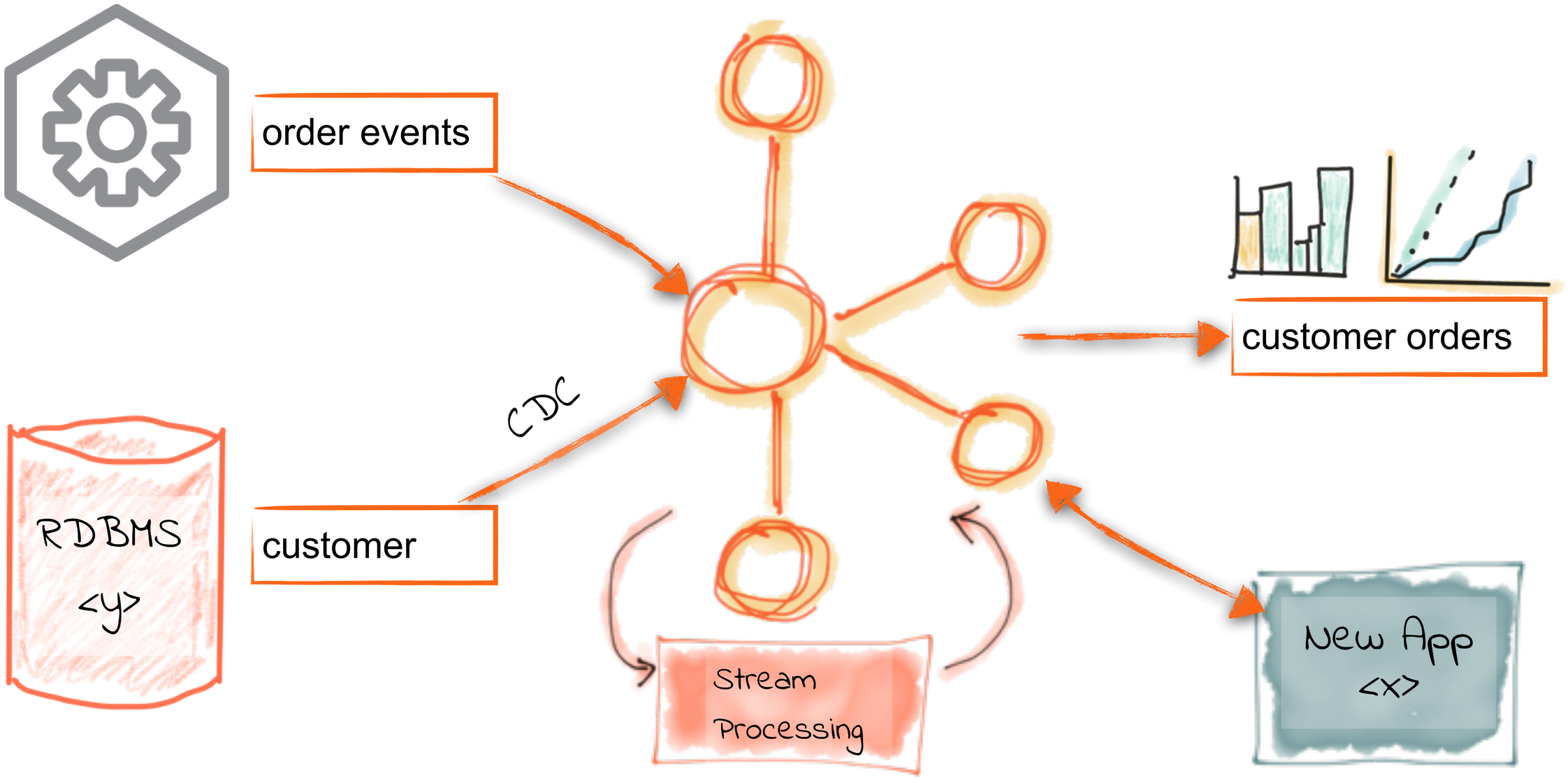
Stream Processing with Apache Kafka and KSQL



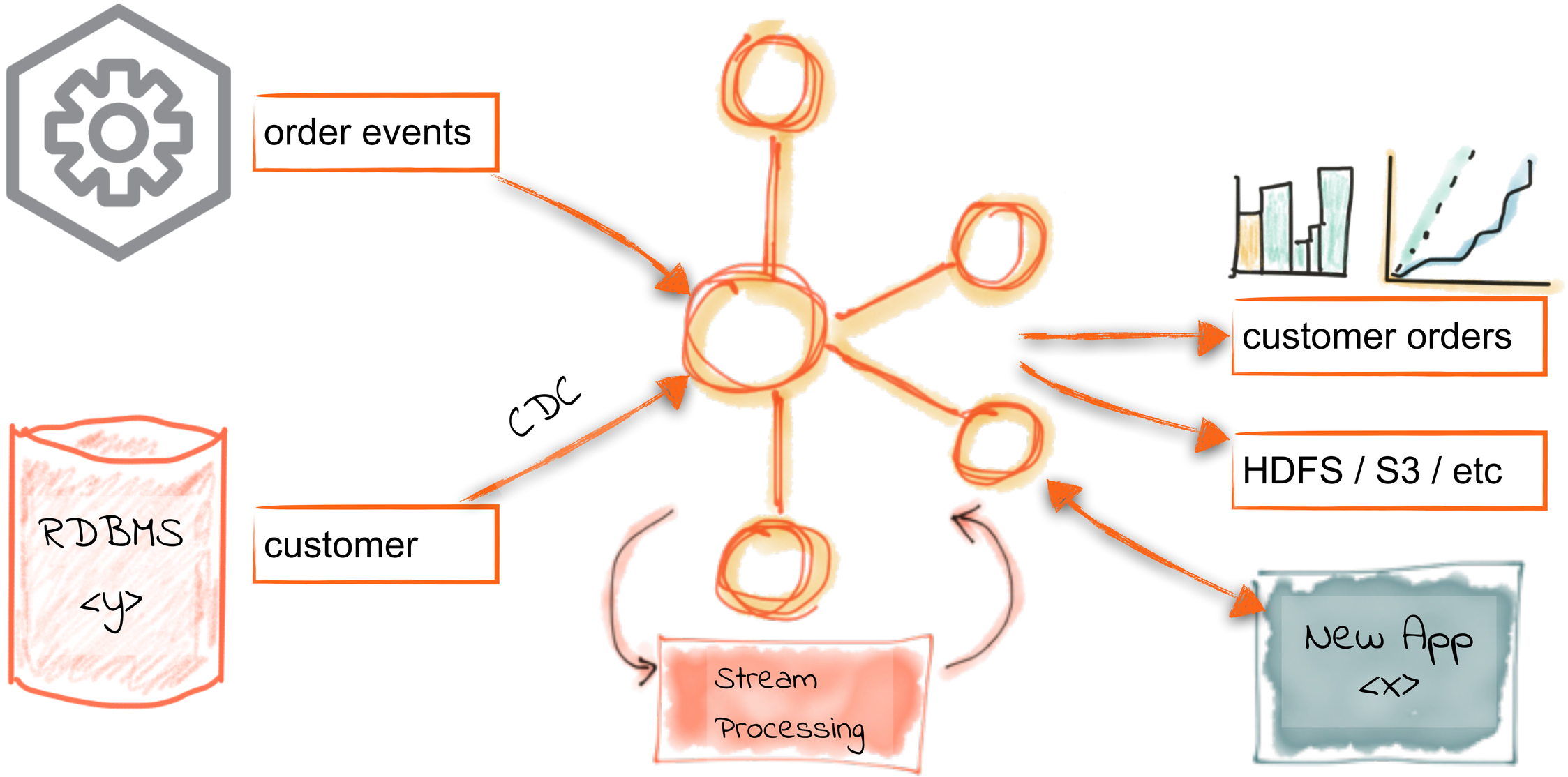
Real-time Event Stream Enrichment



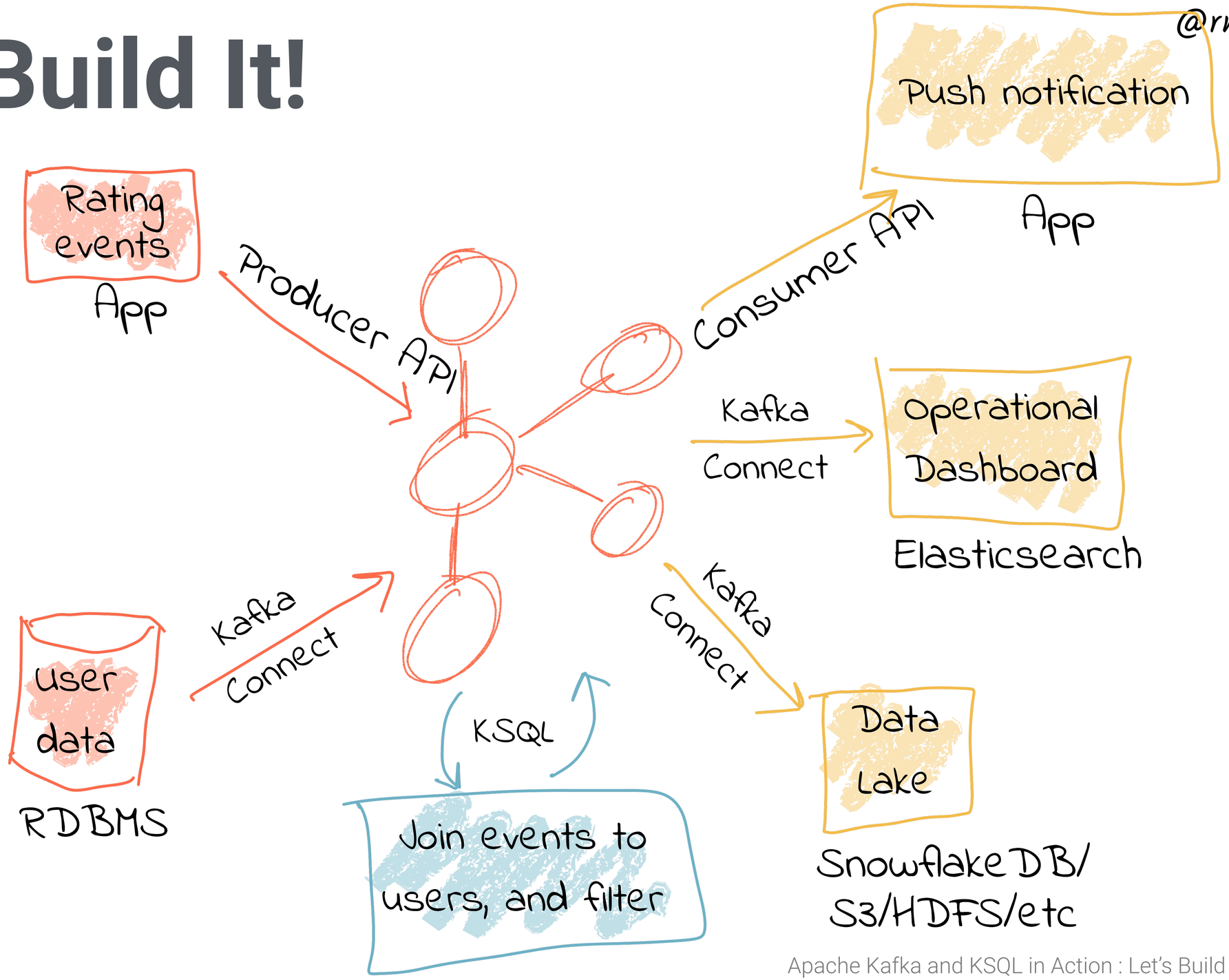
Transform Once, Use Many



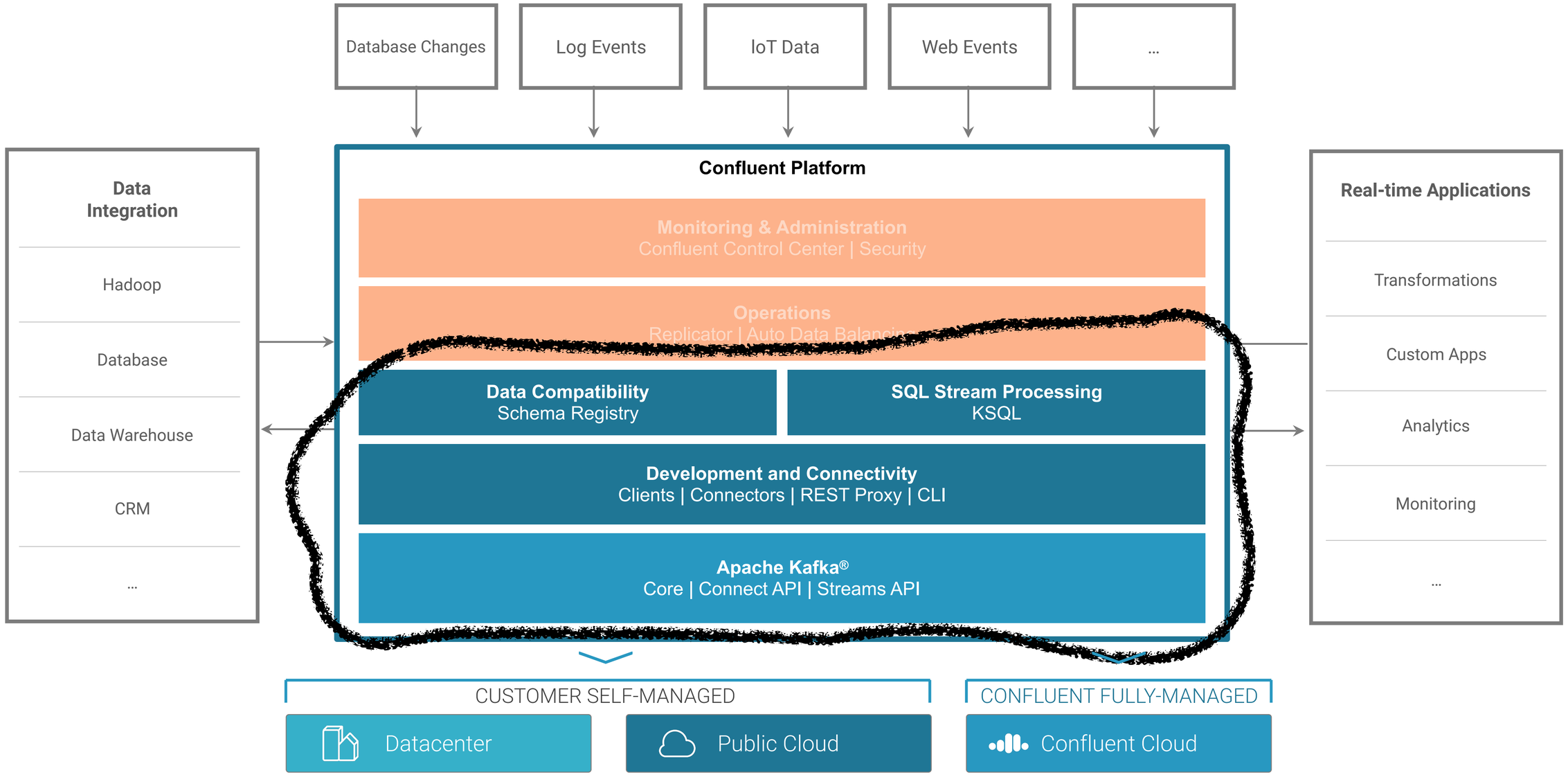
Transform Once, Use Many



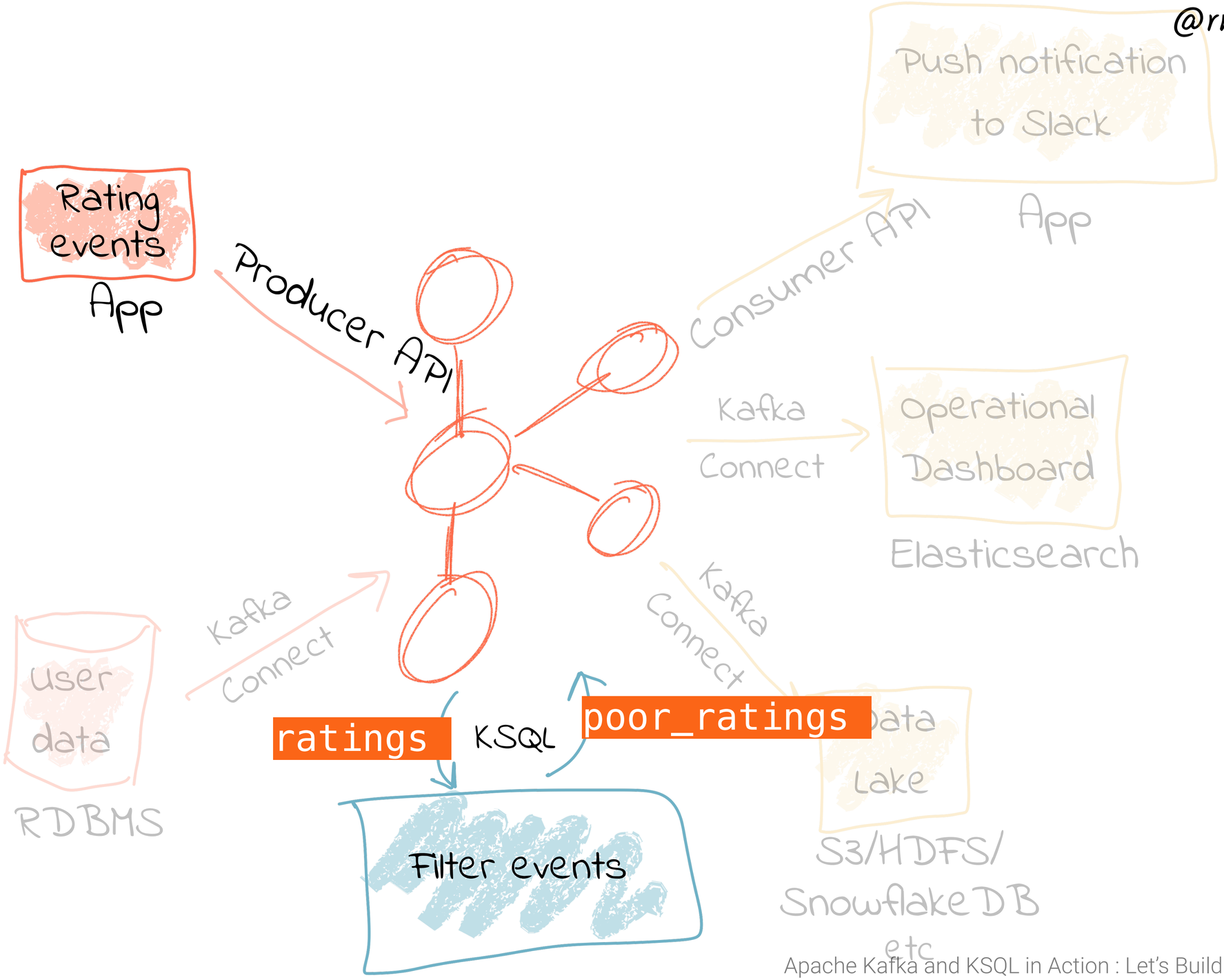
Let's Build It!



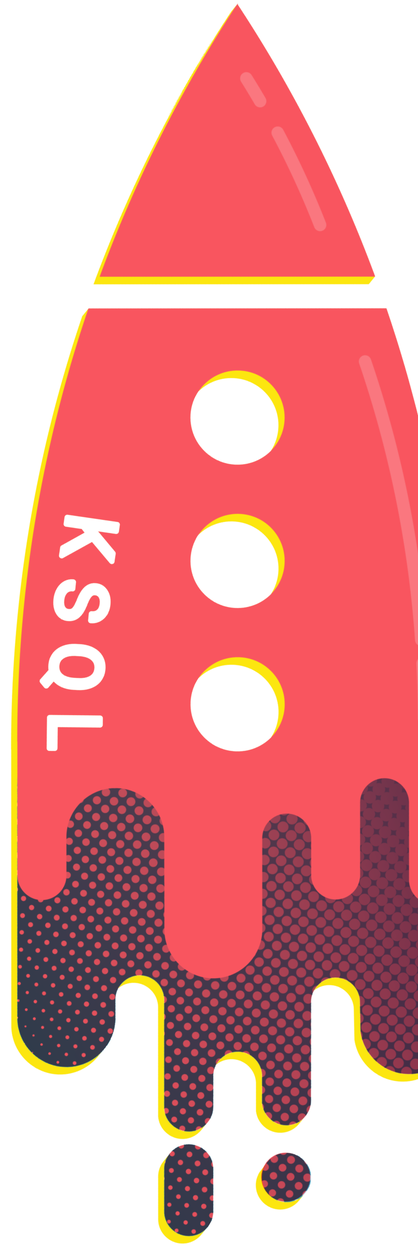
Confluent Community Components
Apache Kafka with a bunch of cool stuff! For free!



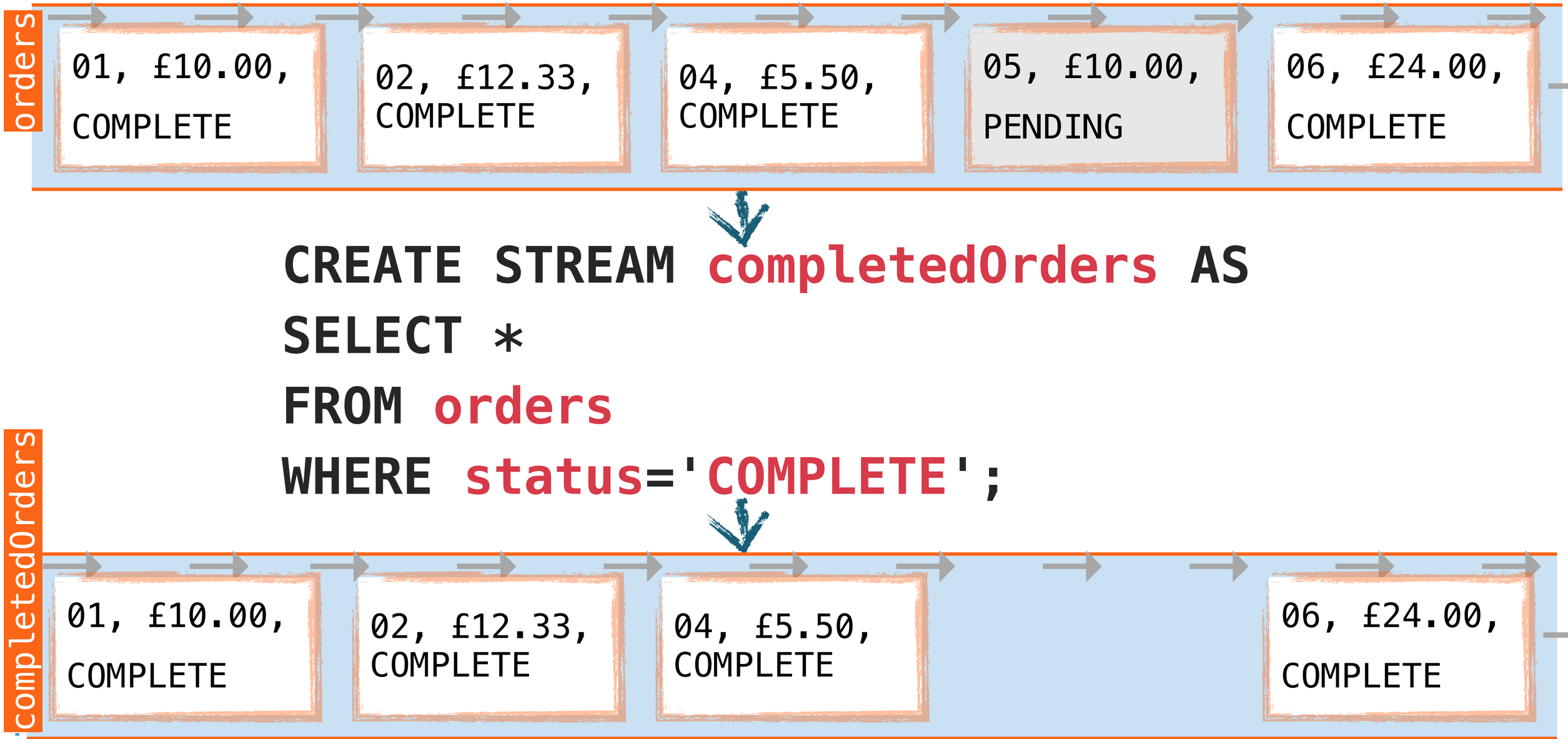
KSQL


Rating events
App
Producer API
Push notification to Slack
App
Consumer API
Kafka Connect
Operational Dashboard
Elasticsearch
Kafka Connect
User data
RDBMS
Kafka Connect
ratings
KSQL
poor_ratings
Filter events
Lake
S3/HDFS/ SnowflakeDB etc

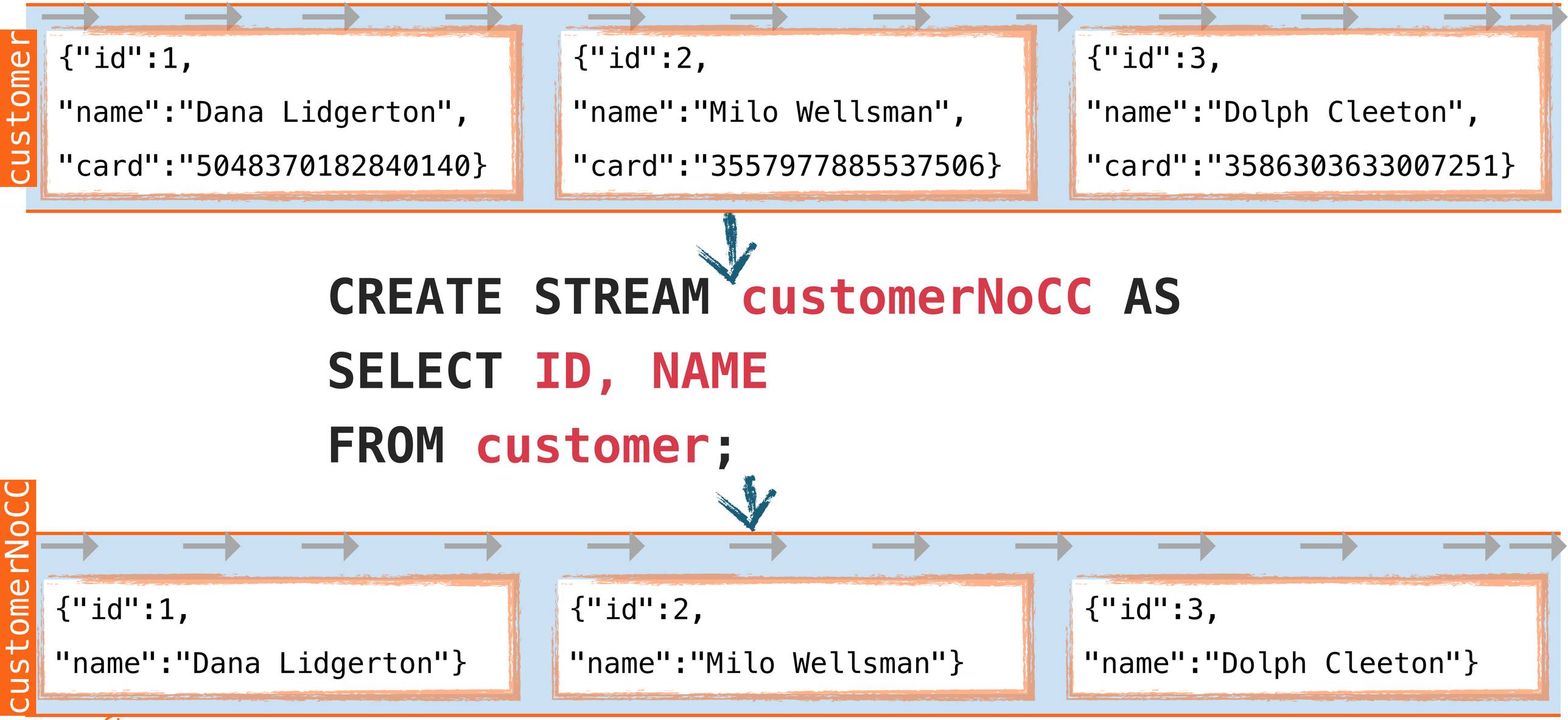
KSQL
is the
Streaming
SQL Engine
for
Apache Kafka



Filter messages with KSQL



Drop columns with KSQL




```
CREATE STREAM customerNoCC AS
SELECT ID, NAME
FROM customer;
```

Stateful aggregation with KSQL

customer

```
{"id":1,
"name":"Dana Lidgerton",
"country":"UK"}
```

```
{"id":2,
"name":"Milo Wellsman",
"country":"UK"}
```

```
{"id":3,
"name":"Dolph Cleeton",
"country":"Germany"}
```

```
CREATE STREAM customersByCountry AS
SELECT country, COUNT(*) AS customerCount
FROM customer WINDOW TUMBLING (SIZE 1 HOUR)
     GROUP BY country;
```

customersByCountry

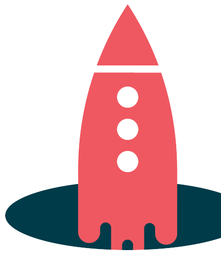
```
{"country":"UK",
"customerCount":2}
```

```
{"country":"Germany",
"customerCount":1}
```

KSQL for Anomaly Detection

Identifying patterns or anomalies in real-time data, surfaced in milliseconds

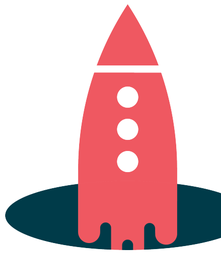
```
CREATE TABLE possible_fraud AS
  SELECT card_number, count(*)
    FROM authorization_attempts
    WINDOW TUMBLING (SIZE 5 SECONDS)
    GROUP BY card_number
    HAVING count(*) > 3;
```



KSQL for Data Transformation

Make simple derivations of existing topics from the command line

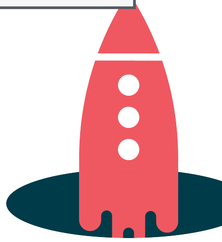
```
CREATE STREAM pageviews
  WITH (PARTITIONS=4,
        VALUE_FORMAT='AVRO') AS
  SELECT * FROM pageviews_json;
```



KSQL for Streaming ETL

Joining, filtering, and aggregating streams of event data

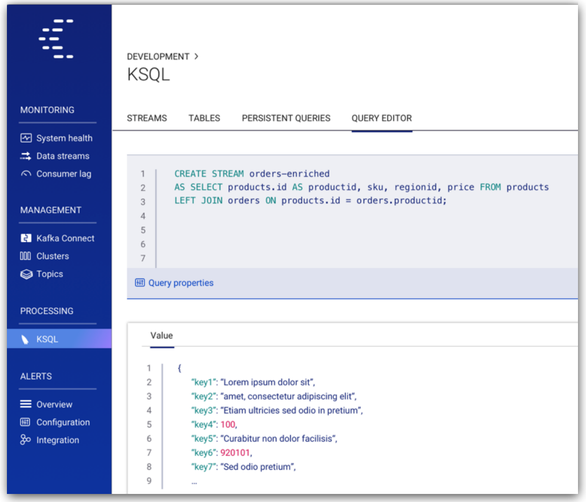
```
CREATE STREAM vip_actions AS
  SELECT userid, page, action
  FROM clickstream c
  LEFT JOIN users u
    ON c.userid = u.user_id
  WHERE u.level = 'Platinum';
```



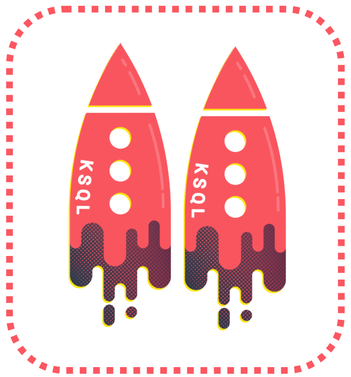
KSQL in Development and Production

Interactive KSQL
for development and testing

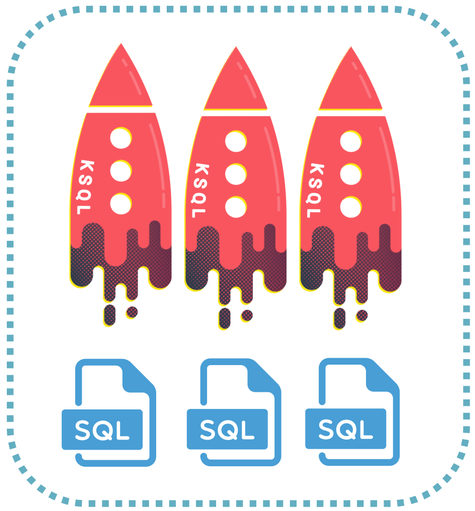
Headless KSQL
for Production



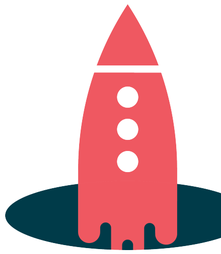
REST



Desired KSQL queries have been identified

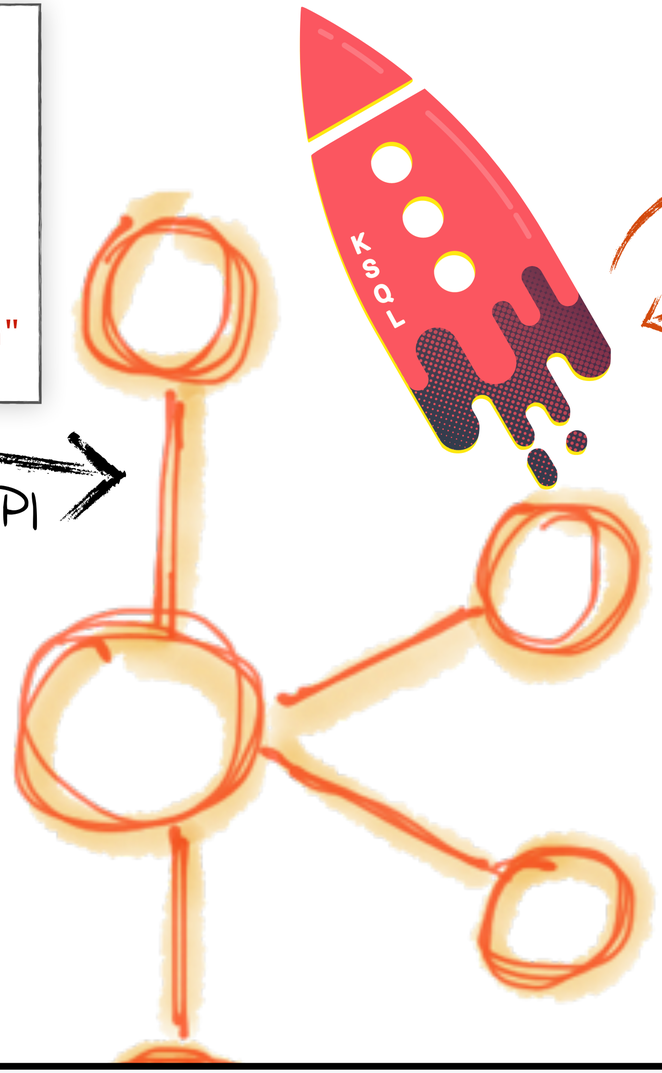


"Hmm, let me try out this idea..."



```
{
 "rating_id": 5313,
 "user_id": 3,
 "stars": 4,
 "route_id": 6975,
 "rating_time": 1519304105213,
 "channel": "web",
 "message": "worst. flight. ever. #neveragain"
}
```

Producer API



Filter all ratings where STARS<3

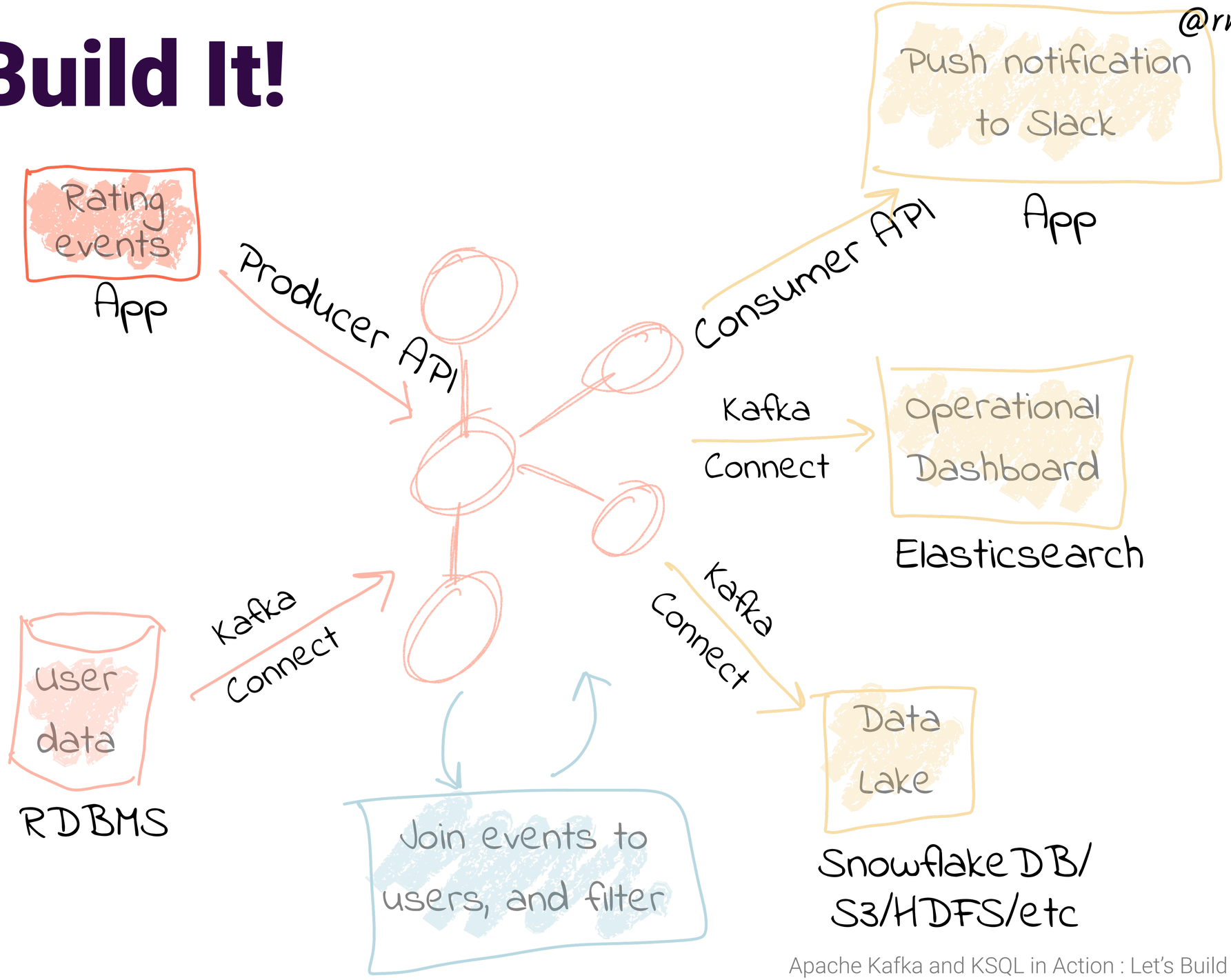
POOR_RATINGS

```
CREATE STREAM POOR_RATINGS AS
SELECT * FROM ratings WHERE STARS <3
```

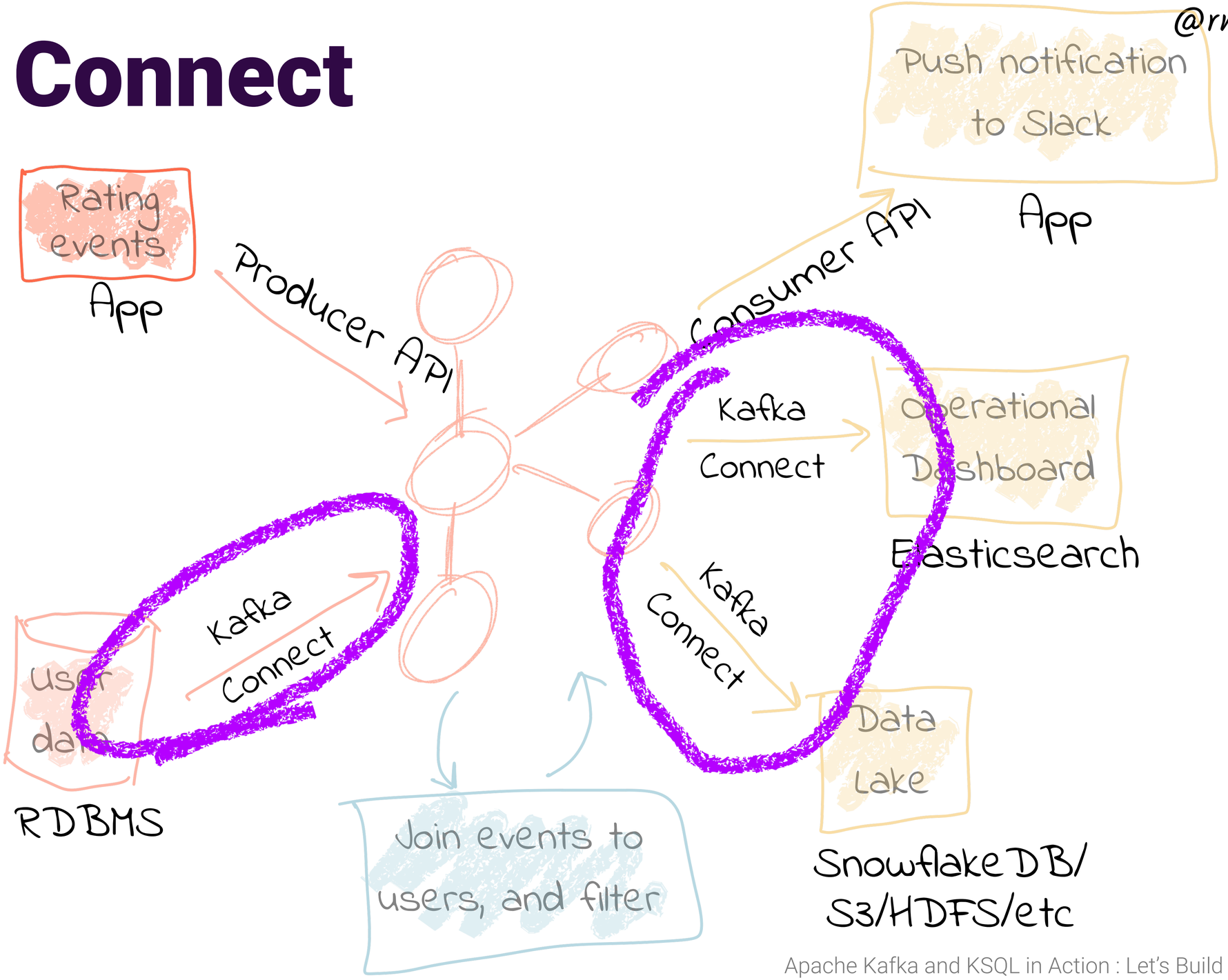
https://cnfl.io/start-ksql-workshop

4. KSQL
5. Querying and filtering streams of data
6. Creating a Kafka topic populated by a filtered stream

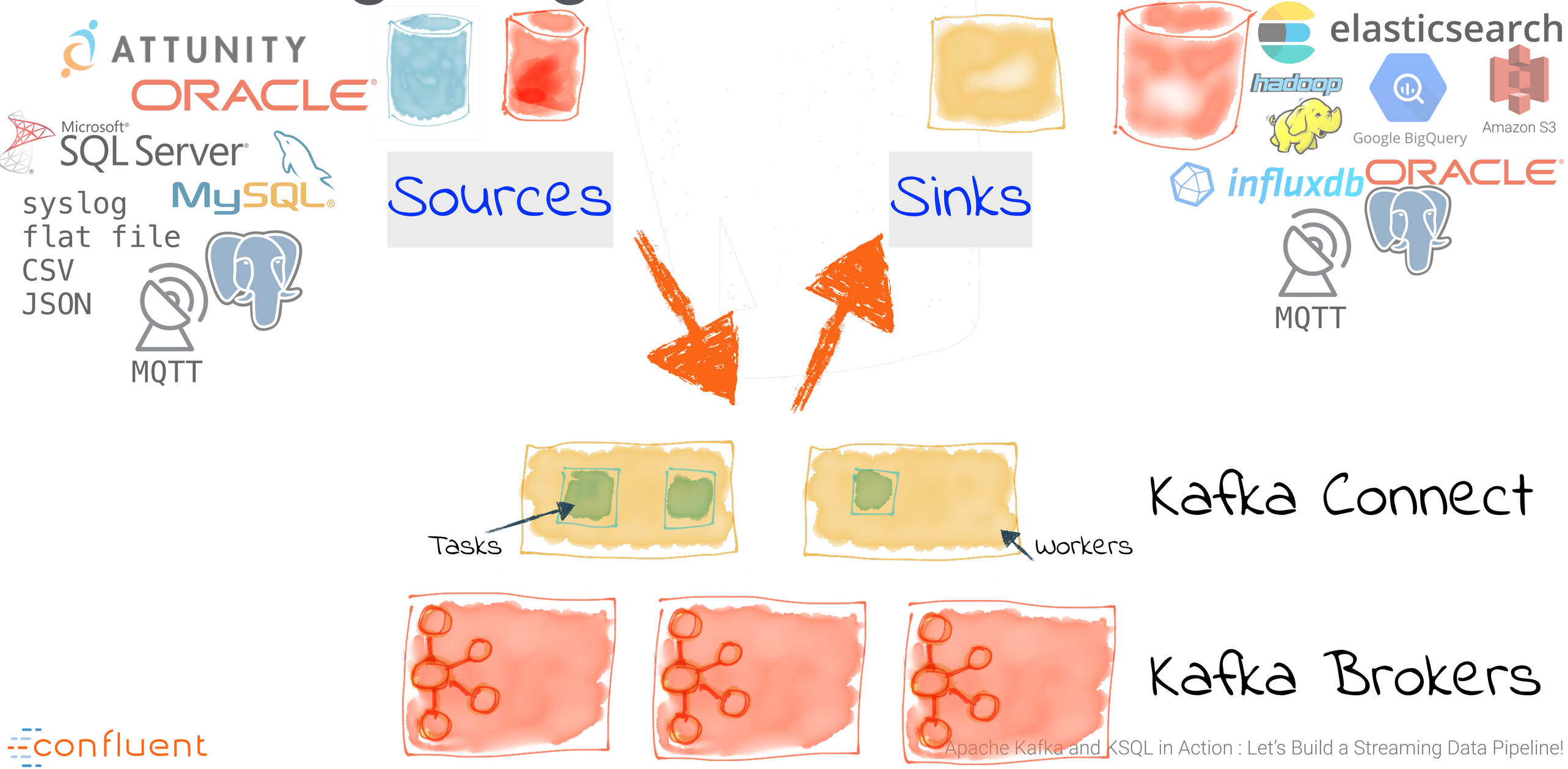
Let's Build It!



Kafka Connect



Streaming Integration with Kafka Connect



Kafka Connect

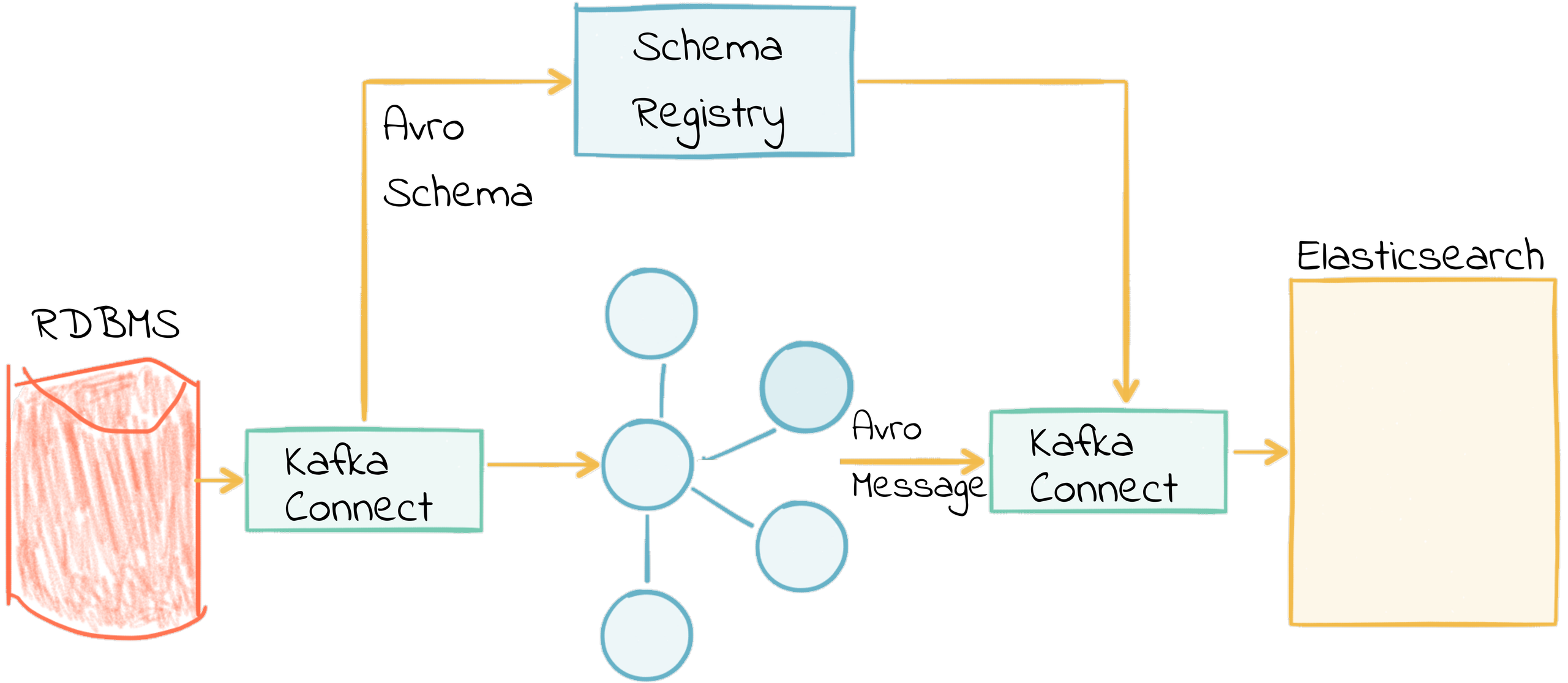
Reliable and scalable integration of Kafka with other systems – no coding required.

- ✓ Centralized management and configuration
- ✓ Support for hundreds of technologies including RDBMS, Elasticsearch, HDFS, S3
- ✓ Supports CDC ingest of events from RDBMS
- ✓ Preserves data schema
- ✓ Fault tolerant and automatically load balanced
- ✓ Extensible API
- ✓ Single Message Transforms
- ✓ Part of Apache Kafka, included in Confluent Open Source

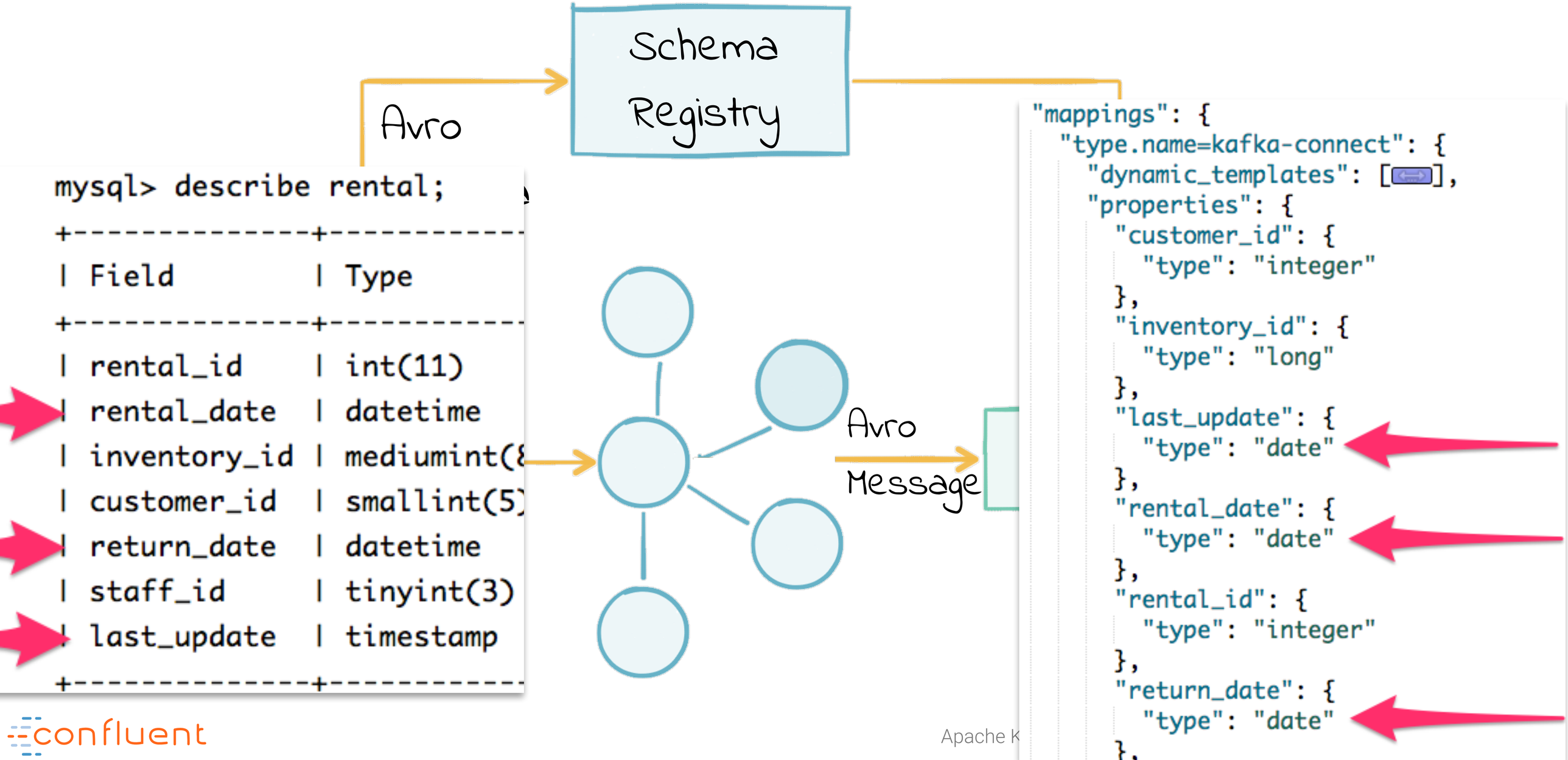
```
{
"connector.class": "io.confluent.connect.jdbc.JdbcSourceConnector",
"connection.url": "jdbc:mysql://localhost:3306/demo?user=rmoff&password=foo",
"table.whitelist": "sales,orders,customers"
}
```

https://docs.confluent.io/current/connect/

Kafka Connect + Schema Registry = WIN

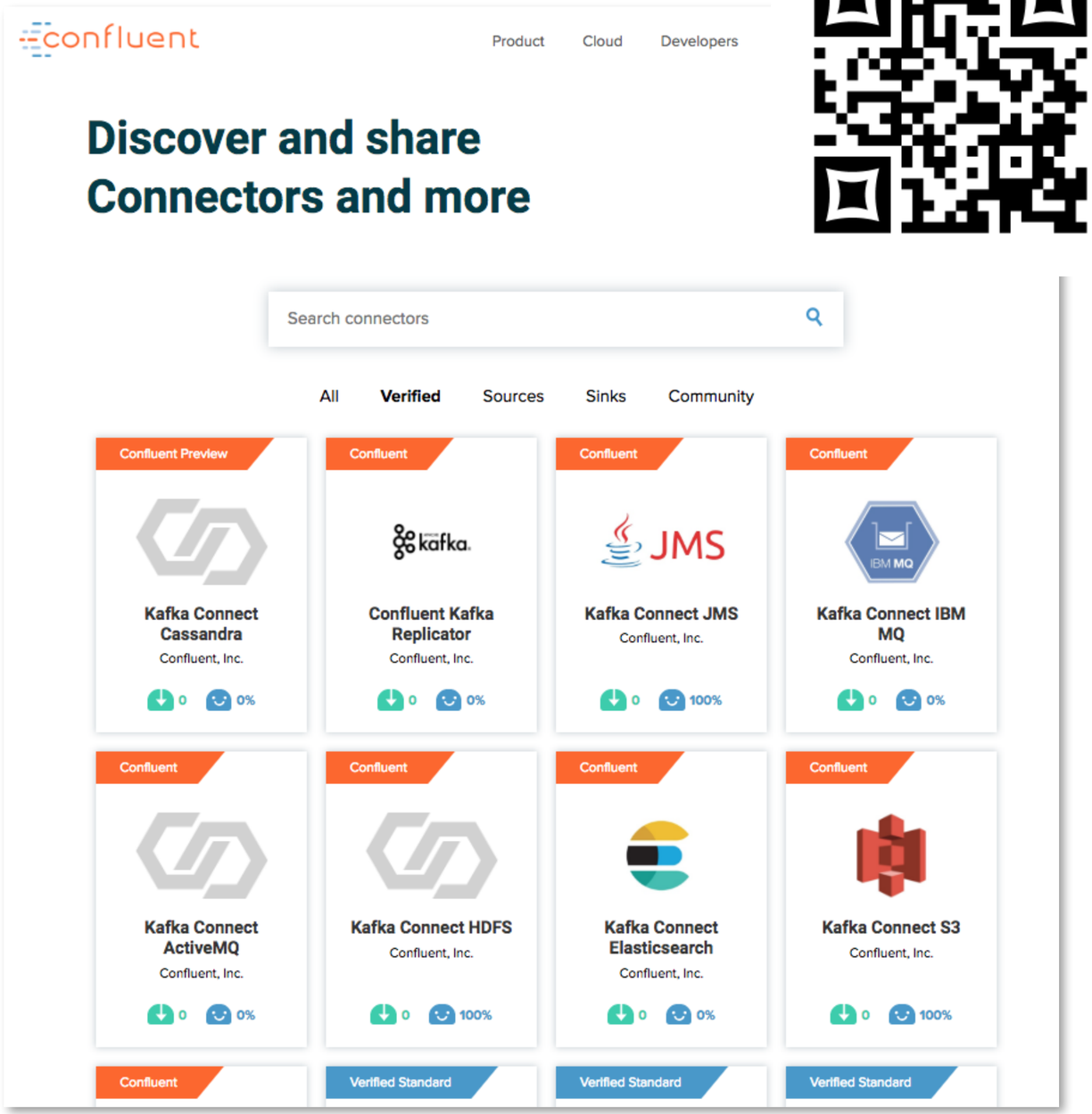


Kafka Connect + Schema Registry = WIN



Confluent Hub

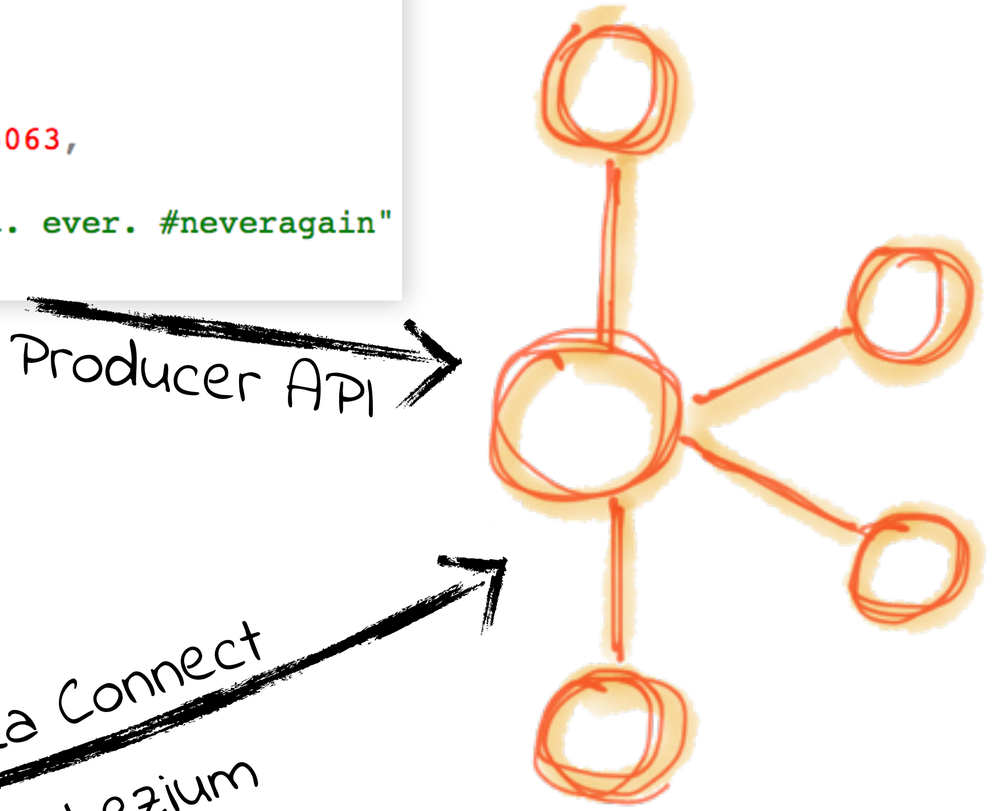
- One-stop place to discover and download :
  - Connectors
  - Transformations
  - Converters



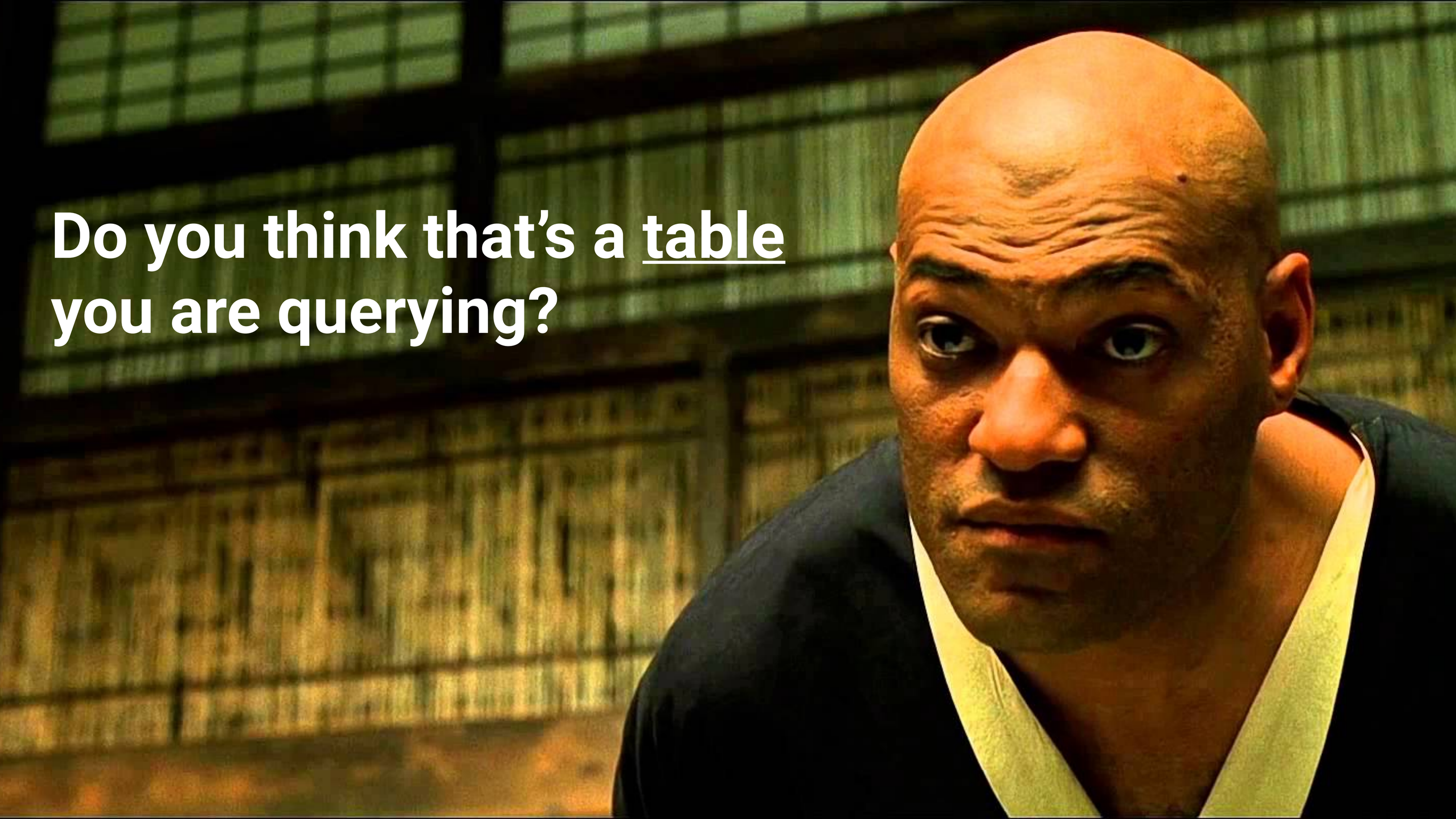
hub.confluent.io

Demo Time!


```
{
  "rating_id": 2133,
  "user_id": 9,
  "stars": 1,
  "route_id": 7219,
  "rating_time": 1519402815063,
  "channel": "web",
  "message": "worst. flight. ever. #neveragain"
}
```


```
{
  "uid": 9,
  "name": "Neha",
  "locale": "en_US",
  "address_city": "Palo Alto",
  "elite": "P"
}
```


Do you think that’s a table
you are querying?

The Stream/Table Duality

Time

Stream

Account ID	Amount
12345	+ €50
12345	+ €25
12345	-€60

Table

Account ID	Balance
12345	€50

Account ID	Balance
12345	€75

Account ID	Balance
12345	€15

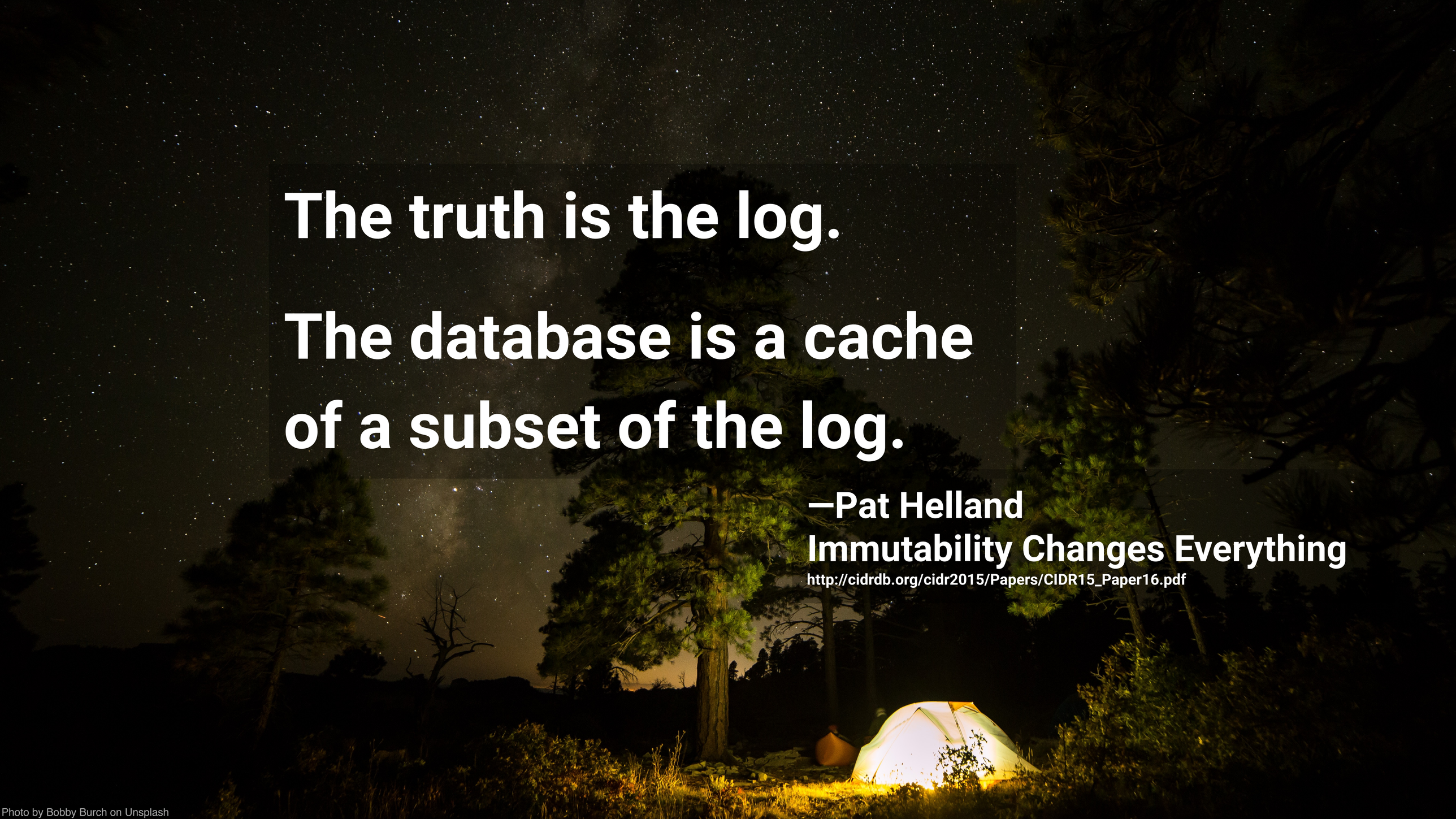
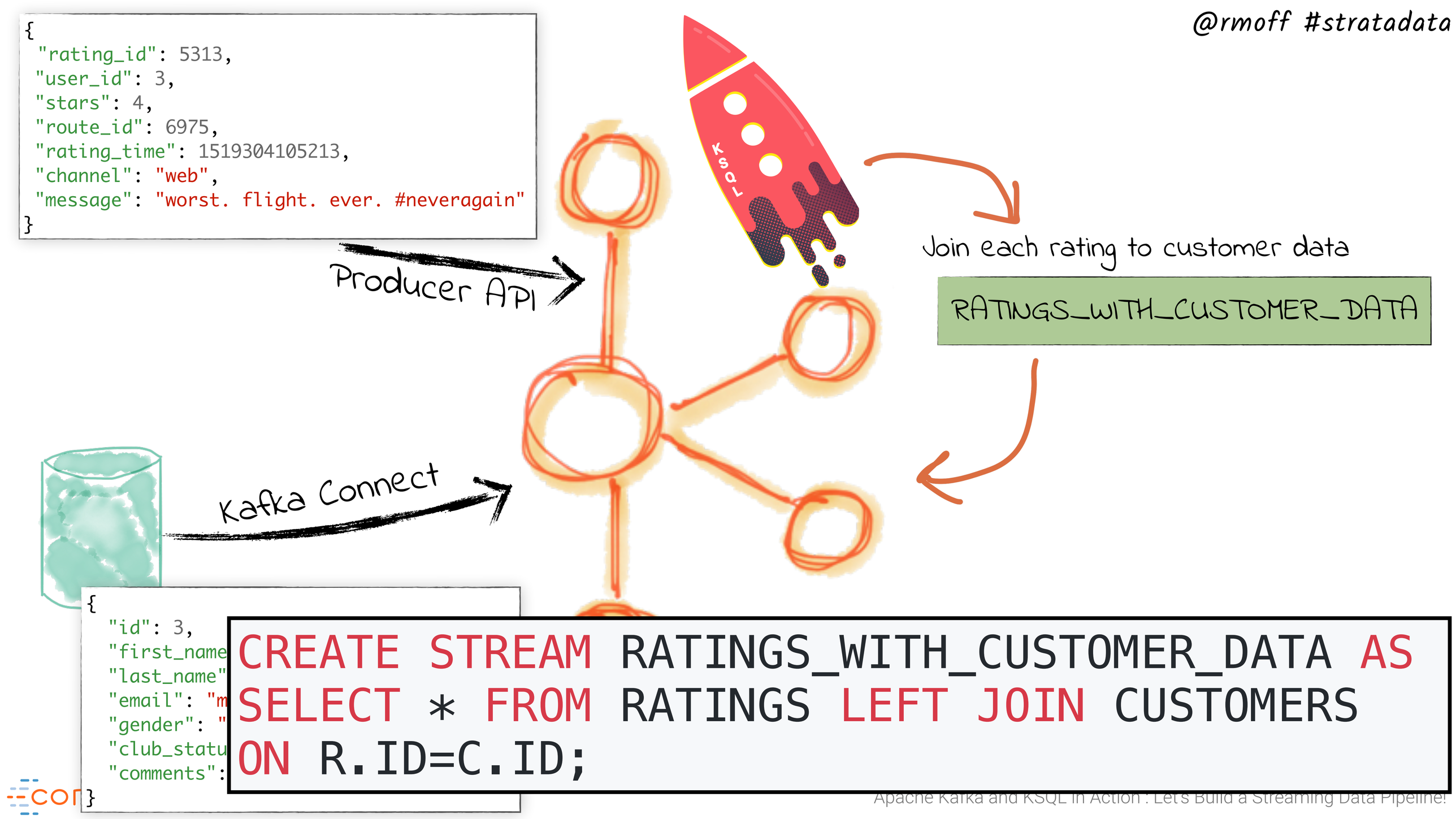
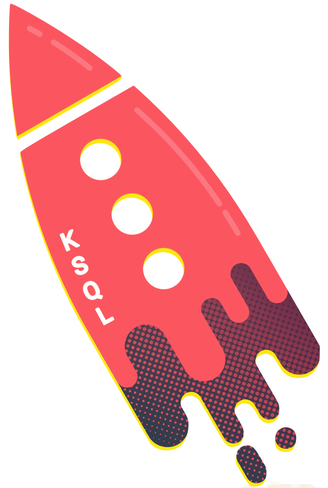


Photo by Bobby Burch on Unsplash


{
"rating_id": 5313,
"user_id": 3,
"stars": 4,
"route_id": 6975,
"rating_time": 1519304105213,
"channel": "web",
"message": "worst. flight. ever. #neveragain"
}
Producer API
KSQL
Join each rating to customer data
RATINGS_WITH_CUSTOMER_DATA
Kafka Connect
{
"id": 3,
"first_name
"last_name"
"email": "m
"gender": "
"club_statu
"comments":
}
CREATE STREAM RATINGS_WITH_CUSTOMER_DATA AS
SELECT * FROM RATINGS LEFT JOIN CUSTOMERS
ON R.ID=C.ID;

```
{
 "rating_id": 5313,
 "user_id": 3,
 "stars": 4,
 "route_id": 6975,
 "rating_time": 1519304105213,
 "channel": "web",
 "message": "worst. flight. ever. #neveragain"
}
```

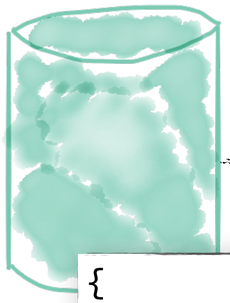

Producer API



Join each rating to customer data

RATINGS_WITH_CUSTOMER_DATA

Kafka Connect



Filter for just PLATINUM customers

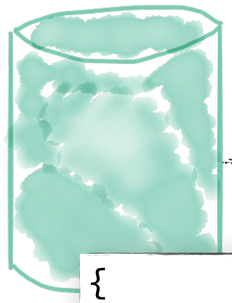
UNHAPPY_PLATINUM_CUSTOMERS


```
{
  "id": 3,
  "first_name
  "last_name"
  "email": "m
  "gender": "
  "club_statu
  "comments":
}
```


```
CREATE STREAM UNHAPPY_PLATINUM_CUSTOMERS AS
SELECT * FROM RATINGS_WITH_CUSTOMER_DATA
WHERE   STARS < 3
```

```
{
 "rating_id": 5313,
 "user_id": 3,
 "stars": 4,
 "route_id": 6975,
 "rating_time": 1519304105
 "channel": "web",
 "message": "worst. flight
}
```


```
CREATE TABLE RATINGS_BY_CLUB_STATUS AS
SELECT CLUB_STATUS, COUNT(*)
FROM RATINGS_WITH_CUSTOMER_DATA
     WINDOW TUMBLING (SIZE 1 MINUTES)
GROUP BY CLUB_STATUS;
```



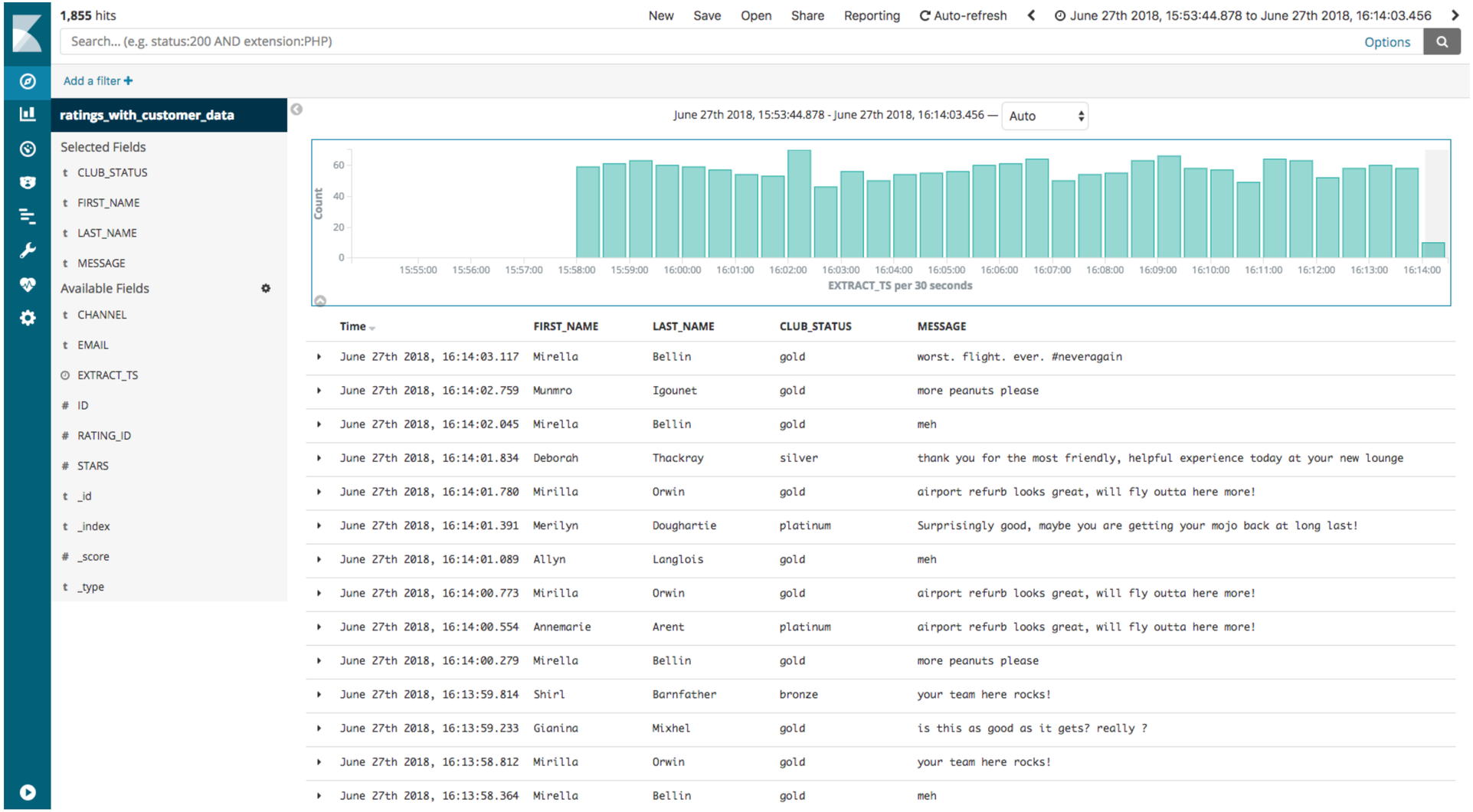
Kafka Connect

```
{
  "id": 3,
  "first_name": "Merilyn",
  "last_name": "Doughartie",
  "email": "mdoughartie1@dedecms.com",
  "gender": "Female",
  "club_status": "platinum",
  "comments": "none"
}
```


Aggregate per-minute by CLUB_STATUS

RATINGS_BY_CLUB_STATUS_1MIN

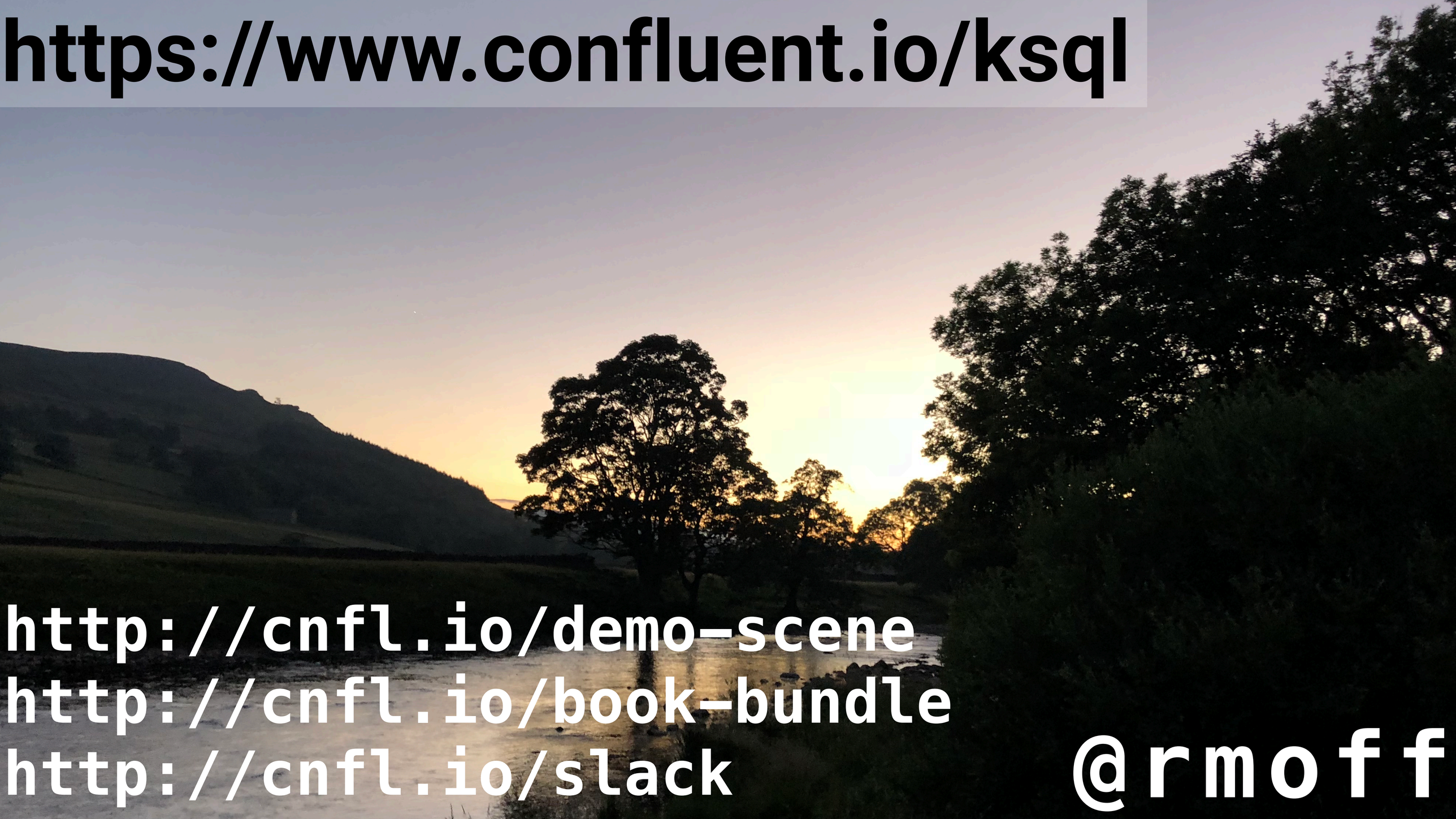
Stream to Elasticsearch



Time	FIRST_NAME	LAST_NAME	CLUB_STATUS	MESSAGE
June 27th 2018, 16:14:03.117	Mirella	Bellin	gold	worst. flight. ever. #neveragain
June 27th 2018, 16:14:02.759	Munmro	Igounet	gold	more peanuts please
June 27th 2018, 16:14:02.045	Mirella	Bellin	gold	meh
June 27th 2018, 16:14:01.834	Deborah	Thackray	silver	thank you for the most friendly, helpful experience today at your new lounge
June 27th 2018, 16:14:01.780	Mirilla	Orwin	gold	airport refurb looks great, will fly outta here more!
June 27th 2018, 16:14:01.391	Merilyn	Doughartie	platinum	Surprisingly good, maybe you are getting your mojo back at long last!
June 27th 2018, 16:14:01.089	Allyn	Langlois	gold	meh
June 27th 2018, 16:14:00.773	Mirilla	Orwin	gold	airport refurb looks great, will fly outta here more!
June 27th 2018, 16:14:00.554	Annemarie	Arent	platinum	airport refurb looks great, will fly outta here more!
June 27th 2018, 16:14:00.279	Mirella	Bellin	gold	more peanuts please
June 27th 2018, 16:13:59.814	Shirl	Barnfather	bronze	your team here rocks!
June 27th 2018, 16:13:59.233	Gianina	Mixhel	gold	is this as good as it gets? really ?
June 27th 2018, 16:13:58.812	Mirilla	Orwin	gold	your team here rocks!
June 27th 2018, 16:13:58.364	Mirella	Bellin	gold	meh

https://cnfl.io/start-ksql-workshop

7. Kafka Connect / Integrating Kafka with a database
8. The Stream/Table duality
9. Joining Data in KSQL
10. Streaming Aggregates
11. Optional: Stream data to Elasticsearch


https://www.confluent.io/ksql
http://cnfl.io/demo-scene
http://cnfl.io/book-bundle
http://cnfl.io/slack
@rmoff

Related Talks

- The Changing Face of ETL: Event-Driven Architectures for Data Engineers
  - 📖 Slides
- ATM Fraud detection with Kafka and KSQL
  - 📖 Slides
  - 👾 Code
  - 📽️ Recording
- Embrace the Anarchy: Apache Kafka's Role in Modern Data Architectures
  - 📖 Slides
  - 📽️ Recording
- Apache Kafka and KSQL in Action : Let's Build a Streaming Data Pipeline!
  - 📖 Slides
  - 👾 Code
  - 📽️ Recording
- No More Silos: Integrating Databases and Apache Kafka
  - 📖 Slides
  - 👾 Code (MySQL)
  - 👾 Code (Oracle)
  - 📽️ Recording

#EOF